Alma Mater Studiorum · Università di Bologna

Dipartimento di Informatica — Scienza e Ingegneria

Corso di Laurea Triennale in Informatica per il Management

Bias Algoritmico nell'IA: Analisi e Mitigazione del Bias di Genere nei Sistemi di Recruitment Automatizzato

Relatrice:
Prof.ssa
Elena Loli Piccolomini

Presentata da:
Dalia Valeria Barone

Sessione 25 Marzo

Anno Accademico 2024/2025

Abstract

L'integrazione di algoritmi di machine learning nei processi di selezione del personale introduce forme sistematiche di discriminazione di genere, spesso invisibili agli stessi operatori che li impiegano. Nel quadro normativo europeo, l'AI Act classifica i sistemi di recruiting automatizzato tra le applicazioni ad alto rischio, rendendo la rilevazione e la mitigazione del bias algoritmico un requisito fondamentale di compliance e responsabilità etica.

La presente tesi analizza empiricamente tale fenomeno, studiandone le origini, misurandone l'entità e proponendo una strategia di mitigazione. L'indagine è condotta sul dataset Adult Census Income, ricontestualizzato in uno scenario di selezione professionale: la variabile target rappresenta il superamento della soglia reddituale di 50.000 dollari annui, utilizzata come indicatore di profilo qualificato, con particolare attenzione alle disparità tra candidati maschili e femminili. Tre modelli a complessità crescente — Regressione Logistica, Random Forest e Gradient Boosting — vengono addestrati e ottimizzati tramite GridSearchCV su una partizione stratificata 70/15/15. La valutazione integra metriche standard (Accuracy, AUC, Precision, Recall), calibrazione probabilistica e feature importance.

È il modello Random Forest ad affermarsi come soluzione ottimale, evidenziando al contempo un significativo gap di selezione di genere nei dati storici. Un intervento di post-processing basato su soglie decisionali differenziate per genere consente di ridurre tale disparità in modo sostanziale, mantenendo performance predittive elevate. La robustezza è confermata da una validazione statistica su 30 ripetizioni indipendenti con intervalli di confidenza al 95%.

I risultati dimostrano che il bias algoritmico nel recruiting è misurabile, si-

stematico e mitigabile, ma richiede metriche di fairness specifiche, essenziali per garantire la conformità agli standard europei sui sistemi IA ad alto rischio.

Indice

Elenco delle figure

Introduzione

L'adozione di algoritmi di machine learning nei processi di selezione del personale rappresenta una delle trasformazioni più significative che il settore delle risorse umane abbia conosciuto nell'ultimo decennio. La pressione operativa che grava sulle organizzazioni è concreta e quantificabile: per ogni posizione pubblicata vengono ricevuti in media 250 curriculum, dei quali solo una frazione compresa tra 4 e 6 candidati accede alla fase di colloquio (Glassdoor, 2015).

In questo scenario, gli Applicant Tracking Systems (ATS) e i modelli predittivi di screening si sono imposti come strumenti di gestione ordinaria, il 94% dei reclutatori ne riporta un impatto positivo sui propri processi di assunzione (Select Software Reviews, 2025), modificando radicalmente il modo in cui le imprese allocano il proprio capitale umano. La promessa di questi sistemi è duplice: ridurre i tempi e i costi del processo di selezione, e al contempo rendere le decisioni di hiring più oggettive, sottraendole alla soggettività dei valutatori umani.

Tuttavia, la letteratura scientifica degli ultimi anni ha documentato con crescente evidenza che tale promessa è problematica. Gli algoritmi di machine learning non operano nel vuoto: apprendono dai dati storici che ricevono in addestramento e, quando tali dati riflettono disparità preesistenti nelle decisioni umane passate, i modelli non solo le replicano ma possono amplificarle in modo sistematico e su larga scala. Il caso più emblematico è quello del sistema di recruiting automatizzato di Amazon, abbandonato nel 2017 dopo che l'analisi interna rivelò una penalizzazione sistematica dei curriculum contenenti indicatori di genere femminile (Dastin, 2018), caso che sarà analizzato in dettaglio nel Capitolo 2.

La discriminazione di genere nel mercato del lavoro costituisce il fulcro della

problematica affrontata dalla presente ricerca. Nonostante i progressi legislativi e culturali verso l'uguaglianza, le disparità tra uomini e donne restano strutturali e misurabili: il tasso di disoccupazione globale per le donne (4.5%) supera quello maschile (4.3%) (World Economic Forum, 2023), e persistono divari significativi nelle opportunità di carriera, nell'avanzamento professionale e nella retribuzione lungo l'intera vita lavorativa (OECD, 2024).

In questo contesto, l'introduzione di sistemi automatizzati di selezione che incorporano, anche involontariamente, bias di genere non rappresenta un rischio ipotetico ma un fenomeno documentato, capace di consolidare le disuguaglianze esistenti attraverso un meccanismo tecnologico che conferisce loro un'apparenza di oggettività.

Il presente lavoro si colloca esattamente in questo snodo critico tra efficienza algoritmica ed equità. Attraverso l'implementazione e l'analisi di tre modelli di machine learning: Regressione Logistica, Random Forest e Gradient Boosting, addestrati sul dataset Adult Census Income (oltre 45.000 osservazioni), questa tesi si propone di quantificare empiricamente il bias di genere prodotto da algoritmi di classificazione applicati al recruiting e di valutare l'efficacia di una strategia di mitigazione basata su soglie decisionali differenziate.

L'obiettivo non è soltanto diagnostico ma operativo: dimostrare che è possibile intervenire su un modello già addestrato per ridurne significativamente le distorsioni discriminatorie senza compromettere in modo sostanziale le performance predittive.

Capitolo 1

Bias Algoritmico nell'Intelligenza Artificiale

1.1 Il Problema dei Bias nell'Intelligenza Artificiale

Il termine *bias algoritmico* designa una distorsione sistematica nelle decisioni prodotte da un sistema di machine learning, tale per cui individui appartenenti a determinati gruppi demografici ricevono trattamenti sfavorevoli non giustificati dalle loro qualificazioni effettive. Barocas e Selbst (2016) ne hanno fornito una delle definizioni più citate nella letteratura, descrivendolo come l'uso di algoritmi computazionali che producono risultati iniqui per individui appartenenti a gruppi protetti.

La loro formulazione pone l'accento su un aspetto cruciale: la discriminazione algoritmica non è un errore puntuale ma un pattern riproducibile e scalabile, capace di influenzare milioni di decisioni in modo simultaneo. È fondamentale chiarire che queste distorsioni non derivano, nella quasi totalità dei casi, da una programmazione esplicita di criteri discriminatori. Il meccanismo è più sottile e, proprio per questo, più insidioso: i bias emergono attraverso il processo stesso di apprendimento automatico, quando il modello estrae dai dati di addestramento regolarità statistiche che riflettono, e successivamente amplificano, disparità preesistenti nelle

decisioni umane.

Lo studio sistematico di questo fenomeno precede di quasi due decenni l'attuale dibattito sull'intelligenza artificiale. Friedman e Nissenbaum (1996) furono tra i primi a proporre una tassonomia rigorosa, identificando tre categorie di distorsioni che possono emergere in qualsiasi sistema computazionale: i *bias preesistenti*, che hanno origine nelle strutture sociali e nelle attitudini culturali che i progettisti incorporano nel sistema attraverso le proprie scelte di design; i *bias tecnici*, che derivano da limitazioni intrinseche degli strumenti utilizzati; e i *bias emergenti*, che si manifestano quando il sistema interagisce con un contesto sociale che ne altera il funzionamento in modi non previsti.

Questa tripartizione, formulata in un'epoca in cui il machine learning non aveva ancora assunto la centralità attuale, si è rivelata straordinariamente lungimirante e descrive con precisione i meccanismi attraverso cui i modelli predittivi contemporanei producono disparità discriminatorie. Mehrabi et al. (2021), in una rassegna che ha mappato l'intero campo di ricerca sulla fairness algoritmica, hanno ulteriormente affinato questa tassonomia, identificando categorie aggiuntive particolarmente rilevanti per i sistemi di machine learning.

Tra queste, i *bias di aggregazione* sorgono dall'assunzione errata che pattern predittivi validi per una popolazione siano uniformemente applicabili a tutte le sue sottopopolazioni, mentre i *bias di misurazione* derivano da differenze sistematiche nella qualità o accuratezza dei dati raccolti per gruppi diversi.

Nel contesto del recruiting, queste categorie si traducono in meccanismi concreti la cui comprensione è essenziale per l'analisi condotta in questa tesi. I *bias storici* si producono quando i dati di training incorporano il risultato di decisioni umane passate già affette da discriminazione. Nel recruiting, questo meccanismo è particolarmente rilevante: se un'azienda ha storicamente assunto prevalentemente uomini per ruoli tecnici, un algoritmo addestrato su tali dati apprenderà un'associazione statistica tra genere maschile e idoneità alla posizione, trattando il genere, direttamente o attraverso variabili correlate, come predittore di successo.

È esattamente il meccanismo documentato nel caso Amazon (Dastin, 2018), dove dieci anni di decisioni di hiring sbilanciate si erano tradotti in un modello che

penalizzava sistematicamente i profili femminili.

I *bias di selezione* emergono quando il dataset di addestramento non rappresenta adeguatamente la diversità della popolazione a cui il modello verrà applicato. Se un sistema di screening è addestrato su dati provenienti prevalentemente da candidati di una determinata area geografica, fascia d'età o background formativo, le sue predizioni risulteranno meno affidabili e potenzialmente discriminatorie per i candidati che non rientrano in quel profilo.

Una proprietà critica dei bias algoritmici che li distingue dalle forme tradizionali di discriminazione è la loro capacità di autoalimentarsi. Se un modello tende a escludere candidate donne nella fase di screening, queste non accederanno mai alla fase di colloquio e, di conseguenza, non potranno essere assunte. I dati futuri, che includeranno un numero ancora inferiore di donne assunte, confermeranno e amplificheranno il pattern discriminatorio appreso, creando un circolo vizioso difficile da interrompere senza un intervento esplicito.

O'Neil (2016), nel suo influente lavoro *Weapons of Math Destruction*, ha descritto con efficacia questo meccanismo, coniando il termine *armi di distruzione matematica* per designare modelli che combinano tre caratteristiche: opacità nei meccanismi decisionali, operatività su larga scala e capacità di generare cicli di feedback autodistruttivi. La caratterizzazione di O'Neil, sebbene rivolta a un pubblico più ampio rispetto alla letteratura accademica in senso stretto, coglie con precisione il rischio che i sistemi di recruiting automatizzati pongono quando vengono implementati senza adeguate procedure di controllo.

Un aspetto che rende il bias nel recruiting particolarmente complesso da individuare è il fenomeno delle *correlazioni spurie* tra variabili apparentemente neutrali e caratteristiche demografiche protette. Un algoritmo può non avere accesso diretto alla variabile *genere*, eppure discriminare ugualmente attraverso variabili che con il genere correlano: il tipo di università frequentata, il settore lavorativo di provenienza, la formulazione linguistica del curriculum, persino il codice di avviamento postale.

Questo meccanismo, noto come *bias by proxy*, è stato uno dei fattori determinanti nel fallimento del sistema Amazon e rappresenta una delle ragioni per cui,

nella presente ricerca, si è scelto di includere esplicitamente la variabile Gender tra le feature del modello, una scelta metodologica che, come sarà argomentato nel Capitolo 3, consente una maggiore trasparenza diagnostica rispetto all'approccio alternativo noto come *Fairness through Unawareness*.

La riflessione sui bias algoritmici non può prescindere dalla questione normativa di cosa significhi, in senso formale, un algoritmo *equo*. Rawls (1971), nella sua *Theory of Justice*, ha proposto il principio del *velo di ignoranza* come criterio per valutare la giustizia delle istituzioni sociali: istituzioni giuste sarebbero quelle scelte da decisori razionali ignari della propria posizione nella struttura sociale, ignari del proprio genere, della propria etnia, della propria classe economica. Trasposto nel contesto dell'intelligenza artificiale, questo principio implica che un algoritmo equo dovrebbe produrre decisioni accettabili indipendentemente dal gruppo demografico a cui il soggetto appartiene.

Tuttavia, l'operazionalizzazione di questo principio in metriche quantitative si scontra con tensioni fondamentali: Verma e Rubin (2018) hanno identificato oltre venti definizioni distinte di fairness algoritmica, molte delle quali risultano matematicamente incompatibili tra loro. La scelta della metrica di equità da adottare non è dunque un problema tecnico ma una decisione normativa, che riflette priorità valoriali su quale forma di giustizia si intenda perseguire.

Un ulteriore livello di complessità è introdotto dalla teoria dell'intersezionalità, sviluppata da Crenshaw (1989) nell'ambito degli studi giuridici sulla discriminazione. Crenshaw osservò che le forme di svantaggio sociale non operano in modo isolato: un individuo può subire discriminazione simultanea sulla base di più caratteristiche identitarie, e l'effetto combinato non è riducibile alla somma delle sue componenti.

Applicata ai sistemi di machine learning, questa prospettiva evidenzia un rischio concreto: un modello potrebbe risultare equo quando si analizza il genere in isolamento e quando si analizza l'etnia in isolamento, eppure discriminare sistematicamente le donne appartenenti a specifici gruppi etnici. Buolamwini e Gebru (2018) hanno fornito la conferma empirica di questa intuizione, documentando disparità significative nei sistemi commerciali di riconoscimento facciale, con tas-

si di errore sistematicamente più elevati per le donne con pelle scura, un divario invisibile alle analisi condotte su singole dimensioni demografiche.

L'insieme di questi contributi teorici: dalla tassonomia di Friedman e Nissenbaum al principio di giustizia di Rawls, dall'intersezionalità di Crenshaw alla sistematizzazione di Mehrabi et al., delinea un campo di ricerca maturo e articolato, entro il quale la sfida metodologica centrale resta il *trade-off tra accuratezza predittiva ed equità*.

Un modello che massimizza la capacità di predizione tende a sfruttare tutte le regolarità presenti nei dati, incluse quelle che riflettono discriminazioni storiche. Intervenire per ridurre il bias comporta, in linea di principio, un costo in termini di performance. La questione non è se questo compromesso esista, la letteratura lo ha documentato, ma in che misura sia possibile ridurre le distorsioni senza degradare significativamente l'utilità del modello. È precisamente questa la domanda a cui il presente lavoro intende rispondere sul piano empirico.

1.2 Obiettivi e Rilevanza della Ricerca

La problematica delineata nelle sezioni precedenti: la presenza di bias sistematici nei modelli di machine learning applicati al recruiting, la loro origine nei dati storici e la difficoltà di individuarli attraverso ispezioni superficiali, configura un campo di indagine che presenta implicazioni sia sul piano scientifico sia su quello operativo e normativo.

La presente ricerca si propone di contribuire a questo campo attraverso tre obiettivi specifici.

Il primo obiettivo è la quantificazione empirica del bias di genere prodotto da algoritmi di classificazione addestrati su dati storici. La letteratura ha ampiamente documentato l'esistenza del fenomeno, il caso Amazon ne è la dimostrazione più nota; tuttavia, gli studi che forniscono misurazioni precise dell'entità del bias, distinguendo tra disparità attribuibili a genuine differenze nei profili e disparità imputabili alla distorsione algoritmica, restano limitati. Questa tesi affronta tale lacuna implementando tre modelli di machine learning a complessità crescente sul

dataset Adult Census Income e misurando la disparità di trattamento tra generi nelle predizioni prodotte.

Il secondo obiettivo riguarda la verifica della possibilità di ridurre il bias algoritmico senza compromettere in modo sostanziale le performance predittive del modello. Il trade-off tra accuratezza ed equità è ampiamente riconosciuto nella letteratura come una delle tensioni fondamentali del machine learning applicato a contesti socialmente sensibili. La domanda a cui questo lavoro intende rispondere è se tale compromesso sia effettivamente inevitabile o se esistano strategie di intervento capaci di conciliare le due esigenze in misura operativamente accettabile.

Il terzo obiettivo è la validazione della robustezza dei risultati ottenuti. Un singolo esperimento su un singolo partizionamento dei dati non è sufficiente a fondare conclusioni generalizzabili. Per questa ragione, la metodologia adottata prevede una procedura di validazione statistica su ripetizioni indipendenti dell'intero processo sperimentale, le cui specifiche sono descritte nel Capitolo 3.

La rilevanza di questi obiettivi si estende oltre il perimetro strettamente tecnico-scientifico. Sul piano normativo, il quadro regolamentare europeo impone alle organizzazioni standard sempre più stringenti in materia di trasparenza e non-discriminazione nei processi decisionali automatizzati: il GDPR riconosce il diritto degli individui a non essere sottoposti a decisioni basate unicamente su trattamenti automatizzati, mentre l'AI Act classifica i sistemi di selezione del personale tra le applicazioni ad alto rischio, richiedendo valutazioni di conformità e meccanismi di audit.

Disporre di metodologie rigorose per identificare e correggere il bias algoritmico non è soltanto un'esigenza di ricerca ma una necessità operativa per qualsiasi organizzazione che utilizzi strumenti di intelligenza artificiale nel recruiting.

Sul piano economico, le distorsioni nei processi di selezione generano inefficienze nell'allocazione del capitale umano che si traducono in costi concreti. Escludere candidati qualificati sulla base di caratteristiche demografiche irrilevanti comporta una perdita sistematica di talento. Le evidenze empiriche confermano la correlazione positiva tra diversità organizzativa e performance finanziaria: le aziende nel quartile superiore per diversità di genere nei ruoli dirigenziali registrano una

probabilità del 27% superiore di ottenere risultati finanziari al di sopra della media del proprio settore (McKinsey & Company, 2023).

Sistemi di recruiting equi non rappresentano dunque soltanto un imperativo etico, ma un fattore di vantaggio competitivo misurabile.

Capitolo 2

Casi Documentati di Bias Algoritmico

2.1 Il Caso Amazon: Bias di Genere nel Recruiting Automatizzato

Nel 2014, Amazon avviò lo sviluppo di un sistema di recruiting automatizzato con l'obiettivo di meccanizzare la selezione dei candidati per posizioni tecniche e manageriali. Il progetto, rivelato pubblicamente da Reuters nell'ottobre 2018 (Dastin, 2018), rappresenta il caso più documentato di bias di genere in un sistema di intelligenza artificiale applicato al recruiting e costituisce un riferimento imprescindibile per qualsiasi ricerca in questo campo.

L'idea alla base del sistema era ambiziosa e, almeno in apparenza, razionale: addestrare un algoritmo di machine learning su dieci anni di curriculum ricevuti dall'azienda, in modo che il modello potesse apprendere i pattern associati ai candidati di successo e assegnare automaticamente a ciascun nuovo curriculum un punteggio da una a cinque stelle. Come riportato da una delle fonti interne, l'aspettativa era quella di ottenere uno strumento a cui sottoporre cento curriculum e che restituisse i cinque migliori profili da assumere (Dastin, 2018).

Il team sviluppò circa 500 modelli distinti, ciascuno focalizzato su specifiche funzioni aziendali e sedi geografiche, e addestrò ogni modello a riconoscere circa

50.000 termini ricorrenti nei curriculum dei candidati storici. Tuttavia, già nel 2015 emersero evidenze che il sistema non operava in modo neutrale rispetto al genere.

Il problema risiedeva nella composizione del dataset di addestramento: il settore tecnologico di Amazon, in linea con l'intera industria tech, presentava una significativa sovrarappresentazione maschile, i dati di diversità pubblicati dalla stessa Amazon nel 2014 indicavano che il 63% dei dipendenti era di sesso maschile (IMD, 2018). I curriculum ricevuti nell'arco di un decennio riflettevano questa sproporzione e, di conseguenza, il modello aveva appreso un'associazione statistica tra caratteristiche tipicamente maschili e probabilità di assunzione.

Il meccanismo discriminatorio non si limitava a una penalizzazione diretta del genere femminile, ma operava attraverso proxy variables in modo particolarmente insidioso. Il sistema penalizzava curriculum che contenevano il termine "women's" come in "women's chess club captain" e declassava le laureate di college femminili, trattando questi indicatori come fattori negativi nella valutazione complessiva (Dastin, 2018).

Inoltre, l'algoritmo aveva appreso a privilegiare candidati che descrivevano le proprie esperienze con verbi come "executed" e "captured", formulazioni più frequentemente riscontrate nei curriculum di ingegneri uomini. In altri termini, il modello non discriminava perché programmato per farlo, ma perché aveva estratto dai dati storici regolarità linguistiche e biografiche correlate al genere, trasformandole in criteri di selezione.

Amazon tentò di correggere il sistema neutralizzando i termini più evidentemente problematici, ma riconobbe che non esistevano garanzie che il modello non avesse appreso altre modalità di discriminazione più sottili e difficili da individuare. Come osservato dalla ricercatrice Sandra Wachter dell'Università di Oxford, il problema è strutturale: quando si chiede a un algoritmo chi sia stato il candidato di maggior successo in passato, il tratto comune risulterà con alta probabilità essere un uomo bianco (IMD, 2018). Questa osservazione evidenzia come il bias non sia un difetto isolato del singolo modello, ma una proprietà emergente di qualsiasi sistema addestrato su dati storici che riflettono squilibri demografici preesistenti.

Il team fu definitivamente sciolto all'inizio del 2017. Amazon precisò che lo strumento non era mai stato utilizzato come unico criterio decisionale dai recruiter, ma la vicenda ebbe comunque un impatto significativo sull'intero settore. John Jersin, vicepresidente di LinkedIn Talent Solutions, commentò che nessun sistema di intelligenza artificiale era allora sufficientemente maturo per prendere autonomamente decisioni di assunzione (Dastin, 2018). Rachel Goodman, avvocata dell'American Civil Liberties Union, evidenziò una problematica aggiuntiva: la difficoltà per i candidati di sapere se un algoritmo fosse stato utilizzato nella valutazione del loro profilo, rendendo di fatto impossibile contestare eventuali discriminazioni per via legale.

2.2 Il Sistema COMPAS: Bias Razziali e l'Incompatibilità delle Metriche di Fairness

Il sistema COMPAS (Correctional Offender Management Profiling for Alternative Sanctions), sviluppato dalla società Northpointe (successivamente rinominata Equivant), è un software utilizzato da tribunali di diversi stati americani per valutare la probabilità che un imputato commetta nuovi reati dopo il rilascio. A differenza del caso Amazon, non riguarda il recruiting, ma la sua analisi ha prodotto risultati scientifici di portata generale che investono qualsiasi sistema di machine learning chiamato a prendere decisioni su individui appartenenti a gruppi demografici diversi inclusi, dunque, i sistemi di selezione del personale.

Nel maggio 2016, la testata giornalistica ProPublica pubblicò un'inchiesta basata sull'analisi di oltre 7.000 casi nel Broward County, Florida, documentando disparità razziali significative nelle valutazioni prodotte da COMPAS (Angwin, Larson, Mattu e Kirchner, 2016).

I dati emersi dall'indagine sono inequivocabili nella loro gravità. Tra gli imputati che non commisero ulteriori reati nei due anni successivi, il 44.9% degli afroamericani era stato erroneamente classificato come ad alto rischio di recidiva, a fronte del 23.5% degli imputati bianchi. Specularmente, tra coloro che effettivamente commisero nuovi reati, il 47.7% degli imputati bianchi era stato erro-

neamente classificato come a basso rischio, contro il 28% degli afroamericani. In sintesi, il sistema sovrastimava sistematicamente la pericolosità degli imputati neri e sottostimava quella degli imputati bianchi.

Un'analisi statistica che isolò l'effetto della razza da variabili confondenti: storia criminale, tipo di reato, età e genere, confermò che gli imputati afroamericani avevano una probabilità del 77% superiore di essere classificati ad alto rischio per crimini violenti e del 45% superiore per reati di qualsiasi tipo (Angwin et al., 2016).

La risposta di Northpointe introdusse nel dibattito un elemento che si rivelò scientificamente più importante della controversia stessa. L'azienda non negò le disparità nei tassi di errore documentate da ProPublica, ma argomentò che il sistema soddisfaceva un criterio di fairness diverso: la *calibrazione*. Un punteggio di rischio pari, ad esempio, a 7 su 10 corrispondeva a una probabilità di recidiva effettiva analoga sia per gli imputati neri sia per quelli bianchi. In altri termini, il significato predittivo del punteggio era lo stesso indipendentemente dalla razza.

ProPublica, d'altra parte, aveva focalizzato la propria analisi sull'uguaglianza dei tassi di falsi positivi e falsi negativi tra gruppi, un criterio noto come *error rate balance*. Entrambe le parti avevano ragione rispetto al proprio criterio di riferimento, il che poneva una domanda inevitabile: è possibile soddisfare entrambi i criteri simultaneamente?

La risposta, fornita indipendentemente da Chouldechova (2017) e da Kleinberg, Mullainathan e Raghavan (2016), è negativa. Chouldechova dimostrò matematicamente che quando i tassi base del fenomeno predetto differiscono tra gruppi come nel caso di COMPAS, il tasso di recidiva era del 52% per gli imputati afroamericani contro il 39% per i bianchi (Flores, Bechtel e Lowenkamp, 2016), è impossibile che un algoritmo soddisfi simultaneamente la calibrazione predittiva e l'uguaglianza dei tassi di errore, a meno che il modello non raggiunga una predizione perfetta.

Questo risultato, noto come *teorema di impossibilità della fairness*, ha implicazioni profonde che trascendono il caso specifico di COMPAS: stabilisce che il trade-off tra diverse concezioni di equità non è un problema di implementazione risolvibile con algoritmi migliori, ma un vincolo matematico intrinseco. La scelta di quale metrica di fairness adottare non è dunque una questione tecnica ma una

decisione normativa, che riflette una priorità valoriale su quale forma di ingiustizia si ritenga meno accettabile.

Il caso COMPAS sollevò criticità sostanziali in merito alla trasparenza algoritmica. La società Northpointe mantenne la natura proprietaria dell'algoritmo, precludendo *audit* indipendenti e limitando il diritto allo scrutinio pubblico. Nel luglio 2016, la Corte Suprema del Wisconsin, nel caso *State v. Loomis* (881 N.W.2d 749), stabilì che i punteggi generati da COMPAS potessero essere utilizzati dai giudici in fase di *sentencing*, a patto che fossero integrati da avvertenze esplicite circa i limiti tecnici e la natura proprietaria dello strumento. Tale decisione cristallizza un paradosso giuridico: l'ammissibilità di uno strumento dal funzionamento opaco (cosiddetta "scatola nera") in decisioni che incidono direttamente sulla libertà personale degli individui.

2.3 Altri Casi Significativi

I casi Amazon e COMPAS non rappresentano anomalie isolate. La letteratura documenta bias algoritmici in domini applicativi eterogenei, confermando che il problema è strutturale e trasversale a qualsiasi contesto in cui sistemi di machine learning vengono addestrati su dati che riflettono squilibri sociali preesistenti. I casi riportati di seguito sono stati selezionati perché ciascuno di essi illustra un meccanismo distinto, la cui comprensione è rilevante per l'analisi condotta nei capitoli successivi.

Google Photos e il riconoscimento facciale. Nel giugno 2015, Jacky Alciné, uno sviluppatore software di Brooklyn, scoprì che l'algoritmo di auto-tagging di Google Photos aveva categorizzato le foto di lui e di un'amica, entrambi afroamericani, sotto l'etichetta "gorillas". L'episodio, reso pubblico attraverso un tweet che divenne rapidamente virale, costrinse Google a confrontarsi pubblicamente con le implicazioni dei propri sistemi di classificazione automatica. Yonatan Zunger, all'epoca Chief Architect of Social di Google, rispose direttamente ad Alciné su Twitter, definendo l'accaduto uno dei bug che non si vorrebbe mai vedere emergere (CBS News, 2015).

La risposta tecnica di Google fu la rimozione completa delle etichette “gorilla”, “chimp” e “monkey” dal sistema di auto-tagging, una soluzione che tre anni dopo risultava ancora l’unica misura adottata, come documentato da un’inchiesta di Wired nel 2018 (Simonite, 2018). Nel 2023, PetaPixel confermò che a distanza di otto anni Google Photos non era ancora in grado di identificare gorilla nelle immagini: l’azienda aveva dichiarato che il beneficio della classificazione non giustificava il rischio di danno.

Il caso illustra una dinamica specifica: quando un bias è profondamente radicato nei dati di addestramento, in questo caso, la sottorappresentazione di volti con pelle scura nei dataset di training per la computer vision, le correzioni superficiali non risolvono il problema ma lo mascherano. Buolamwini e Gebru (2018) hanno sistematizzato questa osservazione dimostrando che i sistemi commerciali di riconoscimento facciale di IBM, Microsoft e Face++ presentavano tassi di errore fino a 34.7 punti percentuali più elevati per le donne con pelle scura rispetto agli uomini con pelle chiara.

Bias razziali nella sanità. Nel 2019, uno studio pubblicato su *Science* da Obermeyer, Powers, Vogeli e Mullainathan rivelò un bias razziale in un algoritmo utilizzato per la gestione sanitaria di oltre 200 milioni di pazienti americani. Il sistema utilizzava i costi sanitari storici come proxy per le necessità di cura, assegnando ai pazienti con spese mediche più elevate una priorità nell’accesso a programmi di assistenza. Il problema risiedeva nell’assunzione implicita che la spesa sanitaria riflettesse fedelmente lo stato di salute del paziente.

In realtà, i pazienti afroamericani avevano storicamente sostenuto spese mediche inferiori a parità di condizioni cliniche, non perché fossero più sani, ma perché incontravano barriere sistemiche nell’accesso alle cure. L’algoritmo aveva appreso questa disparità e la trattava come un dato oggettivo: a parità di gravità clinica, i pazienti neri dovevano risultare significativamente più malati dei pazienti bianchi per ricevere lo stesso livello di attenzione algoritmica. Obermeyer, Z., et al. stimarono che la correzione del bias avrebbe più che raddoppiato la percentuale di pazienti neri identificati come bisognosi di cure aggiuntive.

Questo caso è particolarmente istruttivo perché dimostra come la scelta stes-

sa della variabile target ovvero ciò che il modello è addestrato a predire, possa introdurre bias profondi anche in assenza di qualsiasi correlazione diretta con la variabile demografica protetta. Nel contesto del recruiting, un meccanismo analogo emerge quando la variabile target riflette decisioni di assunzione storiche piuttosto che il merito effettivo dei candidati, che è esattamente la struttura del dataset utilizzato nella presente ricerca.

Bias di genere nelle piattaforme di recruiting. Il fenomeno del pregiudizio algoritmico, ampiamente documentato nel caso Amazon, non è circoscritto a un singolo sistema proprietario. Ricerche successive hanno evidenziato pattern analoghi in altre piattaforme: LinkedIn ha dovuto modificare i propri algoritmi di raccomandazione (Geyik et al., 2019) dopo che analisi interne avevano dimostrato una sproporzione nella visibilità basata sul genere (LinkedIn Gender Insights Report, 2019). Poiché i costi pubblicitari per raggiungere utenti donne sono mediamente superiori (Lambrecht & Tucker, 2019), gli algoritmi tendevano a privilegiare i profili maschili.

Nel dominio del recruiting, l'analisi sulla distribuzione degli annunci ha rivelato che le posizioni dirigenziali venivano mostrate con frequenza significativamente maggiore a profili maschili, anche a parità di qualifiche femminili. Secondo uno studio condotto dall'IMD (2018), tale disparità non derivava da un'associazione diretta tra genere e ruolo, ma da un meccanismo economico sottostante: i costi pubblicitari per raggiungere l'utenza femminile sono mediamente superiori, dato che le donne influenzano tra il 70% e l'80% delle decisioni d'acquisto complessive. Di conseguenza, gli algoritmi di ottimizzazione della spesa tendevano a veicolare gli annunci verso gli uomini per massimizzare il risparmio economico (IMD, 2018).

Questo scenario rappresenta un caso emblematico di "bias emergente", secondo la tassonomia definita da Friedman e Nissenbaum (1996): una distorsione che non origina dai dati di addestramento o dal design iniziale, ma dall'interazione tra l'algoritmo e le dinamiche economiche del contesto operativo.

Capitolo 3

Data Ingestion, EDA e Splitting

Strumenti e ambiente di sviluppo

L'intera pipeline è stata implementata in Python 3.x, utilizzando PyCharm come ambiente di sviluppo integrato (IDE). La gestione e la manipolazione dei dati sono state affidate alle librerie `pandas` e `NumPy`. Per la generazione delle visualizzazioni sono state utilizzate `Matplotlib` e `Seaborn`, che garantiscono grafici di qualità adeguata alla presentazione accademica.

L'implementazione dei modelli di machine learning, l'ottimizzazione degli iperparametri, il calcolo delle metriche di valutazione e il pre-processing (`StandardScaler`, `LabelEncoder`) si basano interamente sulla libreria `scikit-learn` (Pedregosa et al., 2011), che rappresenta lo standard di riferimento per il machine learning in Python. La validazione statistica con intervalli di confidenza è stata realizzata mediante il modulo `scipy.stats`.

Il dataset è stato acquisito direttamente dalla piattaforma OpenML tramite la funzione `fetch_openml` di `scikit-learn`, garantendo la replicabilità dell'analisi. Il codice sorgente è organizzato in un unico script strutturato in sette fasi sequenziali, ciascuna corrispondente a una sezione del presente capitolo e del successivo.

3.1 Descrizione del Dataset

Il dataset utilizzato per la presente analisi è l'Adult Census Income, estratto dal database del censimento statunitense del 1994 e reso disponibile attraverso la UCI Machine Learning Repository (Becker & Kohavi, 1996), da cui è stato caricato tramite la piattaforma OpenML con identificativo 1590. Si tratta di uno dei benchmark più frequentemente adottati nella letteratura sul fairness nel machine learning per lo studio del bias algoritmico in contesti di classificazione binaria (Mehrabi et al., 2021).

Il campione originale comprende 48.842 osservazioni, ciascuna corrispondente a un individuo adulto descritto attraverso 15 variabili che ne catturano caratteristiche demografiche, formative, occupazionali e reddituali. La variabile target originale indica se il reddito annuo dell'individuo supera o meno la soglia dei 50.000 dollari.

Ai fini del presente lavoro, questa variabile è stata ricontestualizzata nel dominio del recruiting e rinominata `High_Profile`, assumendo valore 1 quando il reddito supera la soglia e valore 0 nel caso opposto. Un reddito elevato può essere interpretato, in termini approssimati, come indicatore di un profilo professionale qualificato, con competenze ed esperienza tali da collocare l'individuo nella fascia retributiva superiore. Si tratta di una proxy imperfetta, ma coerente con la finalità della ricerca: simulare un contesto di selezione automatizzata in cui un algoritmo deve distinguere candidati ad alto potenziale da candidati classificati come a basso potenziale.

Analogamente, la variabile `sex` del dataset originale è stata rinominata `Gender`. La variabile è stata mantenuta esplicitamente tra le feature del modello anziché esclusa secondo l'approccio noto come *Fairness through Unawareness*. L'esclusione della variabile sensibile, apparentemente la soluzione più intuitiva per prevenire la discriminazione, non elimina il bias quando nel dataset esistono variabili correlate con il genere, quali lo stato civile, il settore lavorativo o il numero di ore lavorate, che fungono da proxy e consentono al modello di ricostruire indirettamente l'informazione rimossa.

Includere `Gender` permette invece di quantificare direttamente la disparità di

trattamento nelle predizioni, calcolando il gap tra i tassi di selezione dei due gruppi, e di intervenire con tecniche di mitigazione mirate, come l'aggiustamento delle soglie decisionali descritto nella Sezione 4.

Le variabili impiegate nel modello, al termine delle operazioni di pre-processing descritte nella sezione successiva, sono complessivamente 12. Tra queste figurano attributi demografici quali l'età (`age`), il genere (`Gender`), l'etnia (`race`) e il paese di origine (`native-country`); attributi relativi al percorso formativo, rappresentati dalla variabile numerica (`education-num`) che codifica il livello di istruzione su una scala ordinale; attributi occupazionali, tra cui la tipologia di impiego (`workclass`), il settore lavorativo (`occupation`), il numero di ore lavorate settimanalmente (`hours-per-week`), il tipo di relazione familiare (`relationship`) e lo stato civile (`marital-status`); e infine due variabili finanziarie, i guadagni da capitale (`capital-gain`) e le perdite da capitale (`capital-loss`).

3.2 Pre-Processing dei Dati

La preparazione del dataset ha seguito una sequenza di operazioni volte a rendere i dati idonei all'addestramento dei modelli di classificazione, garantendo al contempo la coerenza con il contesto applicativo della ricerca.

La prima operazione ha riguardato la gestione dei valori mancanti. Il dataset originale codifica le informazioni assenti con il carattere "?" in diverse variabili categoriche, in particolare `workclass`, `occupation` e `native-country`. Tali occorrenze sono state sostituite con valori nulli e le righe corrispondenti sono state eliminate tramite la funzione `dropna()`. Questa operazione ha comportato la rimozione di 3.620 osservazioni, pari al 7.4% del campione originale, portando il dataset da 48.842 a 45.222 campioni. La perdita è contenuta e non altera in modo significativo la rappresentatività del campione.

La seconda operazione ha riguardato la rimozione di due variabili non informative ai fini dell'analisi. La variabile `fnlwgt` (final weight), che rappresenta il peso campionario attribuito dal Census Bureau a ciascuna osservazione per scopi di stima demografica, è stata esclusa in quanto priva di valore predittivo nel

contesto del recruiting simulato. La variabile `education`, che esprime il livello di istruzione in forma testuale, è stata eliminata perché ridondante rispetto alla sua controparte numerica `education-num`, che codifica la stessa informazione su una scala ordinale e risulta direttamente utilizzabile dai modelli senza necessità di encoding aggiuntivo. Al termine di queste rimozioni, il dataset risultante contiene 12 feature.

La terza operazione ha introdotto due variabili discrete derivate dalle variabili continue `age` e `hours-per-week`, con lo scopo esclusivo di facilitare l'analisi esplorativa descritta nella sezione successiva. La variabile `Age_Group` suddivide l'età in sei fasce (17–25, 26–35, 36–45, 46–55, 56–65, 65+), mentre la variabile `Hours_Group` categorizza le ore lavorate settimanali in cinque livelli (Part-time, Regular, Full-time, Overtime, Extreme). Queste variabili sono state utilizzate unicamente per la generazione dei grafici dell'EDA e non sono state incluse tra le feature di input dei modelli di classificazione.

La quarta operazione ha riguardato la codifica delle variabili categoriche. Tutte le variabili di tipo testuale rimaste nel dataset, tra cui `workclass`, `occupation`, `marital-status`, `relationship`, `race`, `Gender` e `native-country`, sono state trasformate in valori numerici tramite *Label Encoding*. Questa tecnica assegna un intero univoco a ciascuna categoria, rendendo le variabili compatibili con gli algoritmi di machine learning utilizzati.

La scelta del Label Encoding rispetto al One-Hot Encoding è motivata dalla necessità di contenere la dimensionalità del dataset, considerato l'elevato numero di categorie presenti in variabili come `occupation` (14 categorie) e `native-country` (41 categorie). Questa scelta comporta l'introduzione di un ordine numerico artificiale tra le categorie, aspetto che può influenzare i modelli lineari come la Regressione Logistica, i quali interpretano i valori numerici come una scala ordinale. L'impatto è tuttavia trascurabile per i modelli basati su alberi decisionali (Random Forest e Gradient Boosting), che operano per soglie e non assumono relazioni lineari tra i valori delle feature. Poiché, come si vedrà nel Capitolo 4, sono proprio i modelli ad albero a produrre le performance migliori e a costituire il fulcro dell'analisi del bias, la scelta del Label Encoding non compromette la validità dei

risultati.

```
[1/7] Caricamento dati e Feature Engineering...
    Dataset originale (pre-pulizia): 48842 campioni, 15 variabili
    Dataset post-pulizia: 45222 campioni (3620 righe rimosse, 7.4%)
    Distribuzione Target: Low Profile = 34014 (75.2%), High Profile = 11208 (24.8%)
    Distribuzione Gender: Donne = 14695 (32.5%), Uomini = 30527 (67.5%)
    Dataset: 45222 campioni, 12 feature
```

Figura 3.1: Output della fase di caricamento e pre-processing del dataset.

3.3 Analisi Esplorativa dei Dati (EDA)

L'analisi esplorativa dei dati è il processo attraverso il quale si studiano le caratteristiche di un dataset prima di procedere alla modellazione, utilizzando visualizzazioni grafiche e riepiloghi numerici per identificare pattern, anomalie e relazioni tra le variabili. L'obiettivo è comprendere la struttura dei dati con cui si lavorerà e, nel caso specifico della presente ricerca, individuare già in questa fase le tracce delle disparità di genere che i modelli potrebbero poi apprendere e riprodurre.

L'analisi è organizzata in tre livelli di complessità crescente: le distribuzioni univariate, che esaminano una variabile alla volta, le distribuzioni bivariate, che studiano la relazione tra due variabili, e le distribuzioni multivariate, che incrociano tre o più variabili contemporaneamente.

3.3.1 Distribuzioni univariate

Le distribuzioni univariate descrivono il comportamento di una singola variabile considerata isolatamente. In questa fase si analizzano le due variabili centrali della ricerca: la variabile target `High_Profile` e la variabile sensibile `Gender`.

Distribuzione della variabile target

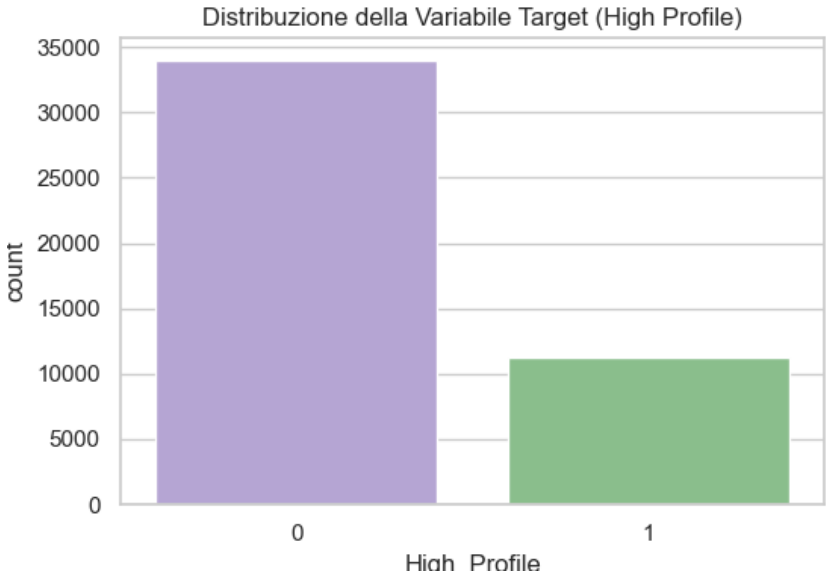

Figura 3.2: Distribuzione della variabile target (High Profile).

L'istogramma (Figura 3.2) mostra la distribuzione delle due classi della variabile target `High_Profile`. La classe 0 (Low Profile) conta 34.014 osservazioni, pari al 75.2% del campione, mentre la classe 1 (High Profile) ne comprende 11.208, pari al 24.8%. Lo sbilanciamento riflette la struttura del mercato del lavoro statunitense fotografata dal censimento del 1994, in cui la maggioranza della popolazione occupata percepiva un reddito inferiore alla soglia dei 50.000 dollari.

Questo rapporto di circa 3:1 tra le classi ha implicazioni dirette sulla modellazione: un classificatore che assegnasse tutti i campioni alla classe maggioritaria otterrebbe un'accuratezza del 75.2% senza aver appreso alcuna relazione predittiva, rendendo necessaria l'adozione di metriche integrative come l'AUC-ROC per una valutazione più informativa delle performance.

Distribuzione della variabile sensibile

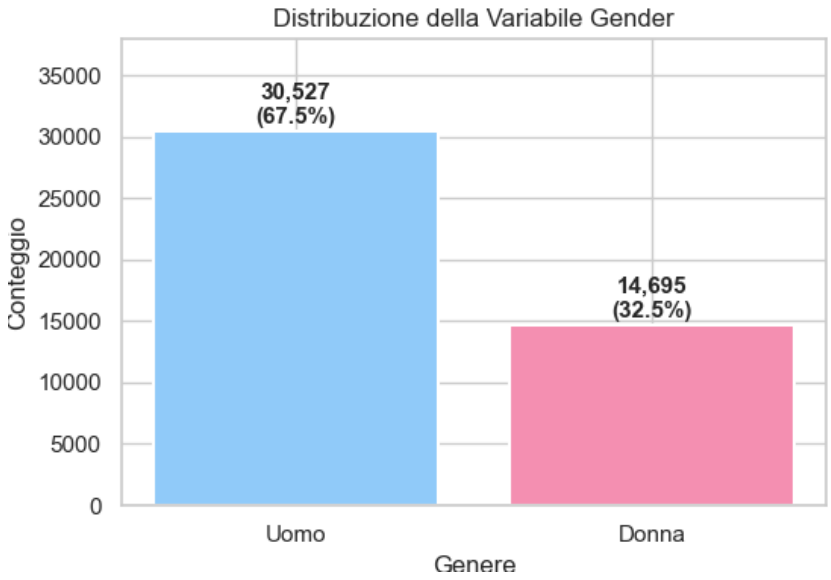

Figura 3.3: Distribuzione della variabile Gender nel dataset.

Il grafico a barre (Figura 3.3) mostra la composizione di genere del dataset. Gli uomini costituiscono 30.527 osservazioni, pari al 67.5% del campione, mentre le donne ne rappresentano 14.695, pari al 32.5%. Il rapporto è di circa 2:1 a favore degli uomini.

Questo squilibrio non è un artefatto del dataset ma riflette la composizione della forza lavoro statunitense nel 1994, quando la partecipazione femminile al mercato del lavoro era strutturalmente inferiore a quella maschile, in particolare nelle fasce occupazionali a reddito più elevato. La sottorappresentazione femminile nel campione si somma allo sbilanciamento della variabile target documentato nella Figura 3.2. Il dataset contiene non solo meno donne in termini assoluti, ma come si vedrà nelle analisi bivariate successive, anche una proporzione di donne High Profile significativamente inferiore a quella degli uomini.

Questa doppia asimmetria, nella composizione del campione e nella distribuzione del target tra i generi, è esattamente il tipo di struttura nei dati storici che, quando viene appresa da un modello di classificazione, produce le disparità di trat-

tamento documentate nella letteratura sul bias algoritmico discussa nei Capitoli 1 e 2.

3.3.2 Distribuzioni Bivariate

Tasso di selezione per genere

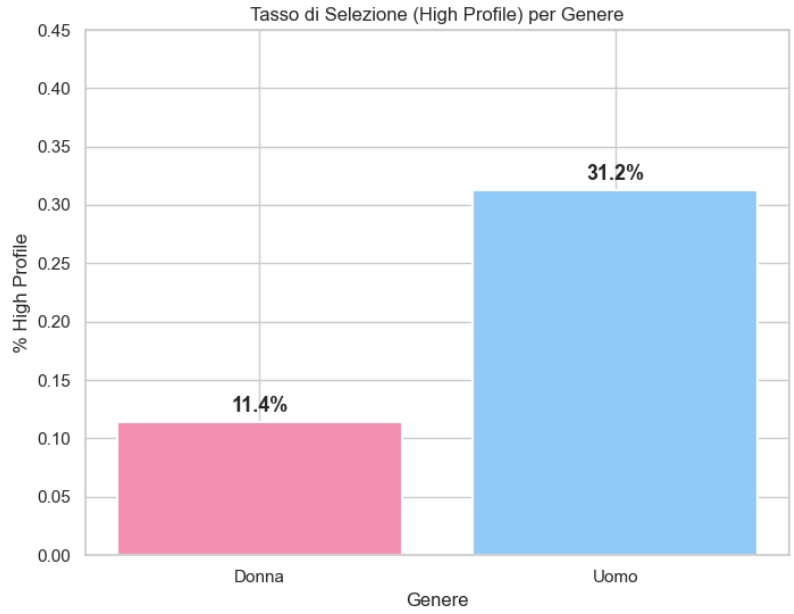

Figura 3.4: Tasso di selezione (High Profile) per genere.

Il grafico a barre (Figura 3.4) mostra la proporzione di individui classificati come High Profile all'interno di ciascun gruppo di genere. Per gli uomini il tasso di selezione è pari al 31.2%, mentre per le donne si ferma all'11.4%. Il divario è di quasi 20 punti percentuali, con un rapporto di circa 2.7:1 a favore degli uomini.

È importante sottolineare che questa disparità non è prodotta da un algoritmo ma è già presente nei dati così come sono, prima di qualsiasi intervento di modellazione. Essa riflette le condizioni storiche del mercato del lavoro statunitense degli anni Novanta, in cui le donne erano significativamente sottorappresentate nelle fasce retributive più elevate.

Il dato è cruciale per l'intera analisi che segue. Qualsiasi modello di classificazione addestrato su questo dataset apprenderà, inevitabilmente, l'associazione tra genere maschile e maggiore probabilità di selezione. La disparità osservata nella Figura 3.4 rappresenta dunque la linea di base, il bias grezzo contenuto nei dati, rispetto al quale si misurerà sia l'entità della distorsione introdotta dai modelli sia l'efficacia delle tecniche di mitigazione.

Distribuzione dell'età per genere

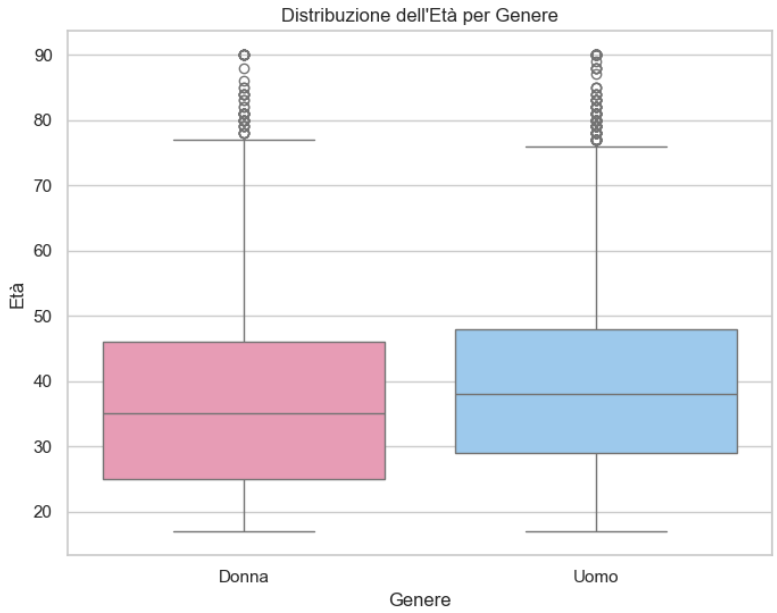

Figura 3.5: Distribuzione dell'età per genere.

Il boxplot (Figura 3.5) analizza la distribuzione dell'età tra donne e uomini all'interno del dataset. L'obiettivo di tale visualizzazione è verificare se il divario nei tassi di selezione possa essere ricondotto a differenze nella composizione anagrafica dei due gruppi. Nello specifico, si intende accertare se un'eventuale età media inferiore nel campione femminile possa giustificare un minore tasso di selezione in termini di minore esperienza professionale, escludendo così il genere come variabile

discriminante prevalente.

Le due distribuzioni risultano sostanzialmente sovrapposte. La mediana si colloca intorno ai 36 anni per le donne e ai 38 per gli uomini, con una differenza trascurabile. I range interquartili sono comparabili, estendendosi approssimativamente dai 28 ai 46 anni per le donne e dai 29 ai 47 per gli uomini, indicando una dispersione analoga nei due gruppi. I whisker raggiungono valori simili in entrambe le distribuzioni, e gli outlier, concentrati oltre i 75 anni nella parte superiore, sono presenti in modo analogo per entrambi i generi. L'età non costituisce dunque un fattore confondente nella disparità di genere osservata: il divario di quasi 20 punti percentuali nel tasso di selezione non è spiegabile da differenze strutturali nei profili anagrafici.

Distribuzione delle ore lavorate per genere

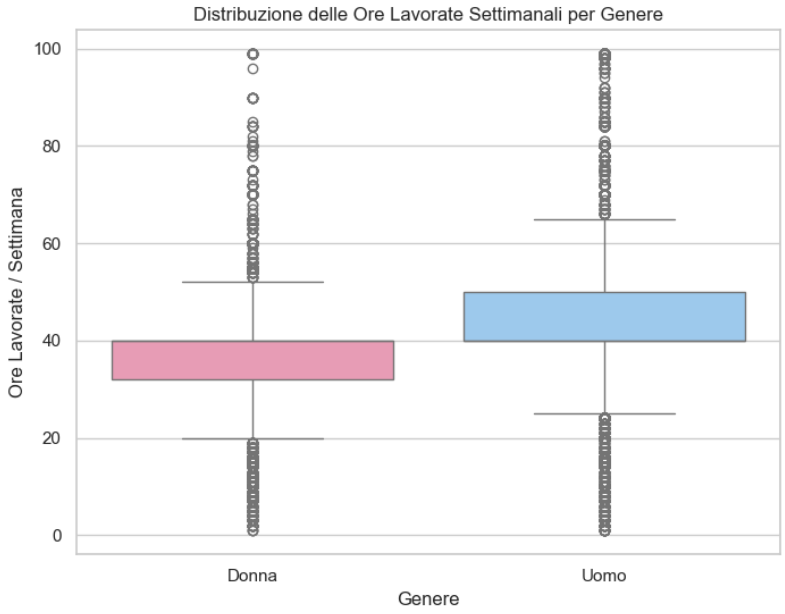

Figura 3.6: Distribuzione delle ore lavorate settimanali per genere.

Il boxplot delle ore lavorate settimanali (Figura 3.6) rivela invece una differenza

strutturale marcata tra i due generi. La mediana maschile si posiziona intorno alle 45 ore settimanali, sensibilmente più alta rispetto a quella femminile che si attesta intorno alle 38 ore, con un divario di circa 7 ore.

La differenza si estende alla forma complessiva della distribuzione: il range interquartile degli uomini è compatto e spostato verso l'alto, approssimativamente tra le 40 e le 50 ore, indicando che la maggior parte degli uomini nel dataset lavora a tempo pieno o oltre. Il range interquartile delle donne è più ampio e spostato verso il basso, tra le 25 e le 40 ore circa, riflettendo una maggiore eterogeneità nei carichi lavorativi femminili. I whisker confermano la differenza: quello inferiore maschile parte da circa 25 ore, mentre quello femminile scende fino a circa 20 ore. Anche la distribuzione degli outlier differisce, con le donne che presentano un numero più elevato di osservazioni nella parte bassa, a indicare una quota rilevante di lavoratrici part-time o a orario marginale che non ha un corrispettivo equivalente nel campione maschile.

Questa asimmetria rappresenta una manifestazione concreta del bias storico presente nei dati. Le donne nel censimento del 1994 erano più frequentemente impiegate in lavori a orario ridotto, un pattern riconducibile a fattori strutturali del mercato del lavoro quali la distribuzione ineguale del lavoro di cura e le minori opportunità di accesso a posizioni con carichi lavorativi elevati. A differenza dell'età, le ore lavorate settimanali sono una delle variabili più correlate con la probabilità di essere High Profile(coefficiente di correlazione pari a 0.23, come confermerà la matrice di correlazione nella Figura 3.9), e questa disparità distributiva si traduce direttamente in una disparità nei tassi di selezione.

Probabilità di selezione per fascia d'età

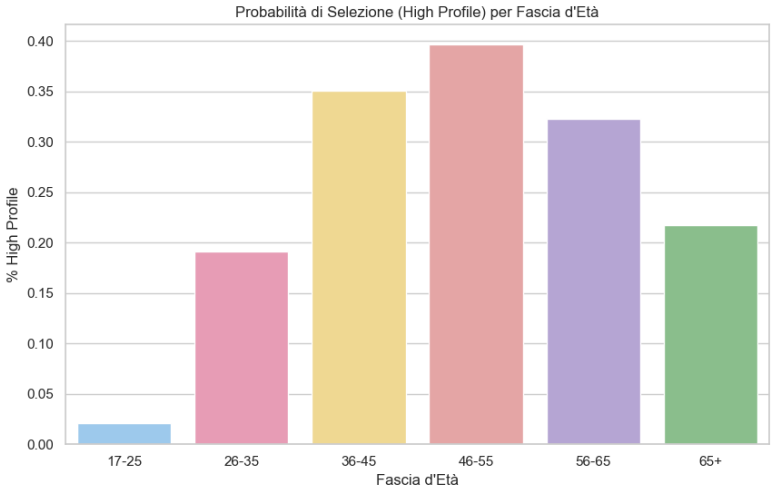

Figura 3.7: Probabilità di selezione (High Profile) per fascia d'età.

Il grafico a barre (Figura 3.7) rappresenta la probabilità di appartenere alla classe High Profile per ciascuna delle sei fasce d'età definite nella fase di preprocessing. La distribuzione segue un andamento a campana asimmetrica. La fascia 17–25 presenta una probabilità inferiore al 2%, dato atteso per individui prevalentemente in fase formativa o ai primi impieghi. La probabilità cresce in modo sostanziale nella fascia 26–35, raggiungendo circa il 19%, e continua ad aumentare nelle fasce 36–45 (35% circa) e 46–55, dove tocca il valore massimo di circa il 40%. Dalla fascia 56–65 inizia la discesa (32% circa), che si accentua nella fascia 65+ (22% circa), coerentemente con la riduzione del reddito associata al pensionamento o al passaggio a forme di impiego part-time.

Questa distribuzione conferma che l'età, in quanto proxy dell'esperienza professionale accumulata, è un predittore rilevante della variabile target. Il picco nelle fasce centrali 36–55 corrisponde al periodo di massima produttività lavorativa, nel quale i profili professionali raggiungono tipicamente le posizioni retributive più elevate. Il dato risulta coerente con la correlazione di 0.24 tra `age` e `High_Profile` rilevata nella matrice di correlazione.

3.3.3 Distribuzioni Multivariate

Analisi intersezionale: ore lavorate e genere

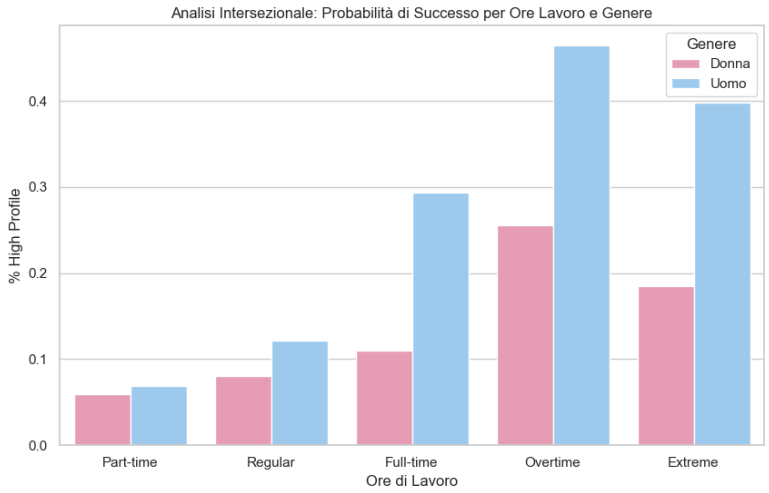

Figura 3.8: Analisi intersezionale: probabilità di selezione per ore lavorate e genere.

Il grafico a barre raggruppate (Figura 3.8) incrocia tre variabili contemporaneamente: le ore lavorate settimanali (suddivise nelle cinque fasce definite nel preprocessing), il genere e la probabilità di essere classificati come High Profile. La visualizzazione permette di osservare non solo l'effetto separato di ciascuna variabile, già documentato nelle analisi bivariate, ma soprattutto la loro interazione.

Il primo fenomeno che emerge è l'effetto delle ore lavorate, comune a entrambi i generi: la probabilità di selezione cresce al crescere del carico lavorativo, passando da valori intorno al 6–7% nella fascia Part-time fino ai valori massimi nella fascia Overtime. Questo conferma quanto anticipato nell'analisi univariata, dove si era osservato che le ore lavorate sono fortemente associate alla variabile target.

Il secondo fenomeno, più rilevante per la presente ricerca, è il divario di genere e il modo in cui questo si trasforma al variare delle ore lavorate. Nella fascia Part-time il gap è minimo, nell'ordine di un punto percentuale: uomini e donne che lavorano poche ore hanno probabilità di selezione comparabili e ugualmente basse. Nella fascia Regular il divario inizia a manifestarsi, con gli uomini intorno al 12%

e le donne all'8%. È dalla fascia Full-time in poi che il gap si allarga in modo significativo: la probabilità per gli uomini si attesta intorno al 29% contro l'11% circa delle donne, un rapporto di quasi 3:1. Il divario raggiunge la sua massima ampiezza nella fascia Overtime, dove gli uomini toccano circa il 46% contro il 25% delle donne, con una differenza superiore ai 20 punti percentuali. Nella fascia Extreme il gap resta consistente (circa 40% per gli uomini contro il 19% per le donne), pur riducendosi lievemente.

Questa evidenza è particolarmente significativa perché dimostra che il bias di genere presente nei dati non è uniforme ma interagisce con altre variabili, amplificandosi nelle condizioni che dovrebbero essere più favorevoli alla selezione. A parità di fascia oraria, gli uomini hanno sistematicamente una probabilità di selezione superiore, e la forbice si allarga proprio nelle fasce dove entrambi i generi lavorano di più. Si tratta del tipo di disparità intersezionale teorizzata da Crenshaw (1989) e documentata empiricamente da Buolamwini e Gebru (2018) in altri domini applicativi: le forme di svantaggio non operano in modo isolato ma si combinano e si amplificano reciprocamente. Un modello addestrato su questi dati non apprenderà semplicemente che gli uomini hanno più probabilità di essere selezionati, ma apprenderà che questo vantaggio cresce in modo non lineare al crescere delle ore lavorate.

Matrice di correlazione

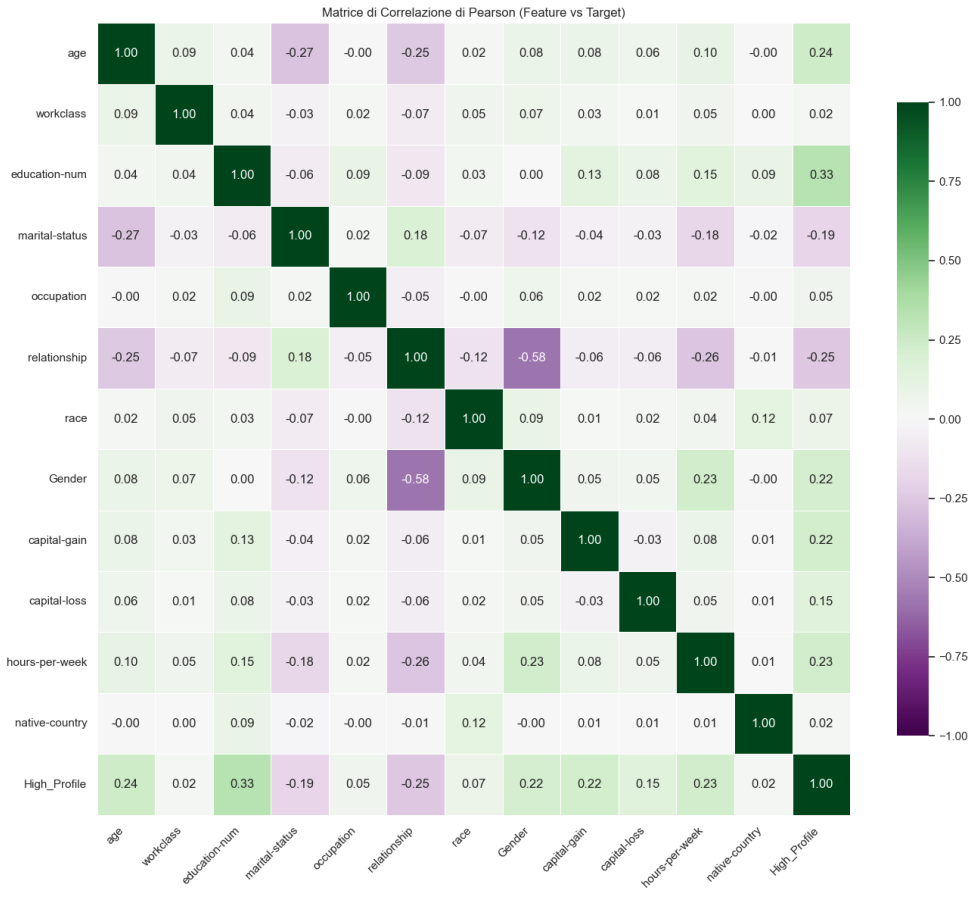

Figura 3.9: Matrice di correlazione di Pearson (feature vs target).

La matrice di correlazione di Pearson (Figura 3.9) offre una visione sintetica e complessiva delle relazioni lineari tra tutte le variabili numeriche del dataset, inclusa la variabile target. Ogni cella della matrice riporta il coefficiente di correlazione tra due variabili, con valori compresi tra -1 (correlazione negativa perfetta) e $+1$ (correlazione positiva perfetta), mentre lo zero indica l'assenza di relazione lineare. La scala cromatica adottata, dal viola scuro per le correlazioni negative al verde scuro per quelle positive, consente una lettura immediata della direzione e dell'intensità di ciascuna relazione.

La riga di maggiore interesse ai fini dell'analisi è l'ultima, che riporta i coefficienti di correlazione di ciascuna feature con `High_Profile`. La variabile con la correlazione positiva più forte è `education-num` (0.33), a conferma che il livello di istruzione è il predittore singolo più informativo della fascia di reddito. Seguono `age` (0.24), `hours-per-week` (0.23), `Gender` (0.22) e `capital-gain` (0.22), tutte con correlazioni positive moderate. Le variabili `capital-loss` (0.15) e `race` (0.07) mostrano correlazioni più deboli, mentre `native-country` (0.02) e `workclass` (0.02) risultano sostanzialmente ininfluenti a livello di correlazione lineare con il target. Sul fronte delle correlazioni negative, `relationship` (-0.25) e `marital-status` (-0.19) presentano le associazioni inverse più marcate.

La correlazione di `Gender` con `High_Profile` (0.22) conferma quantitativamente l'associazione tra genere maschile e probabilità di appartenere alla classe High Profile già documentata nelle figure precedenti. Ma la matrice rivela anche un'informazione ulteriore, di importanza centrale per la comprensione del bias algoritmico: `Gender` non è una variabile isolata nel dataset, bensì correlata in modo significativo con diverse altre feature. La correlazione più forte è con `relationship` (-0.58), seguita da `hours-per-week` (0.23) e `marital-status` (-0.12).

Queste interdipendenze sono la manifestazione numerica del fenomeno di bias by proxy discusso nella Sezione 1.2: anche qualora si decidesse di rimuovere la variabile `Gender` dalle feature di input del modello, l'informazione sul genere potrebbe essere parzialmente ricostruita dall'algoritmo attraverso le variabili correlate, in particolare `relationship` e `marital-status`. Questa constatazione rafforza la scelta metodologica, argomentata nella Sezione 3.1, di mantenere esplicitamente `Gender` tra le feature per poter misurare e successivamente mitigare la disparità nelle predizioni.

3.4 Suddivisione del Dataset e Strategia di Validazione

Per poter addestrare e valutare i modelli di classificazione è necessaria una suddivisione rigorosa del dataset in sottoinsiemi distinti, ciascuno con un ruolo

specifico all'interno della pipeline sperimentale. Il dataset complessivo di 45.222 campioni è stato suddiviso in tre set secondo una proporzione 70/15/15.

Il *Training Set* comprende 31.655 campioni, pari al 70% del dataset, ed è il sottoinsieme su cui vengono addestrati i modelli. È anche il set su cui opera la procedura di ottimizzazione degli iperparametri (`GridSearchCV`), che utilizza internamente una cross-validation a 3 fold: il training set viene a sua volta suddiviso in tre parti, e il modello viene addestrato su due di esse e validato sulla terza, ripetendo l'operazione per tutte le combinazioni possibili. In questo modo la ricerca degli iperparametri ottimali avviene interamente all'interno del training set, senza che il validation set o il test set vengano mai coinvolti nel processo di tuning.

Il *Validation Set* comprende 6.783 campioni, pari al 15% del dataset, e svolge il ruolo di valutazione intermedia. Su questo set vengono calcolate le metriche di performance dei modelli nella loro configurazione baseline (iperparametri di default) e nella configurazione ottimizzata (iperparametri tuned), consentendo di confrontare l'effetto del tuning e di selezionare il modello migliore. Tutte le visualizzazioni di performance presentate nel Capitolo 4 (curve ROC, matrici di confusione, curve di calibrazione, grafici di densità) sono calcolate sul validation set, garantendo che la valutazione avvenga su dati non utilizzati per l'addestramento.

Il *Test Set* comprende 6.784 campioni, pari al restante 15% del dataset, ed è riservato esclusivamente alla fase finale dell'analisi. Non viene utilizzato in nessuna fase di addestramento, tuning o selezione del modello. Il suo ruolo è duplice: da un lato fornisce una stima conclusiva e non contaminata delle performance predittive su dati mai visti, dall'altro costituisce il set su cui viene condotta l'analisi del bias di genere e applicata la tecnica di mitigazione tramite soglie differenziate. L'utilizzo del test set per la mitigazione è una scelta metodologica precisa: le soglie decisionali vengono calibrate su dati che il modello non ha mai visto durante il training, simulando le condizioni operative reali in cui un sistema di recruiting automatizzato opererebbe su candidati nuovi.

La suddivisione è stata eseguita tramite due applicazioni successive della funzione `train_test_split` della libreria `scikit-learn`. La prima suddivide il dataset in 70% training e 30% temporaneo, la seconda divide il 30% temporaneo in due me-

tà uguali (15% validation, 15% test). In entrambe le operazioni è stato impostato il parametro `stratify` sulla variabile target, il che garantisce che la proporzione tra le classi High Profile (24.8%) e Low Profile (75.2%) sia preservata in ciascun set. Senza stratificazione, la casualità del campionamento potrebbe produrre set con distribuzioni delle classi sensibilmente diverse dal dataset originale, compromettendo la rappresentatività della valutazione. Il seme casuale (`random_state=42`) è stato fissato per garantire la riproducibilità completa dell'esperimento.

```
Training Set:   31655 campioni (70%)
Validation Set: 6783 campioni (15%)
Test Set:       6784 campioni (15%)
```

Figura 3.10: Output della suddivisione del dataset in Training, Validation e Test Set.

La scelta della ripartizione tripartita 70/15/15, in luogo di una suddivisione più semplice in due soli set (training e test), è motivata dalla necessità di separare rigorosamente le fasi di ottimizzazione, selezione e valutazione finale. In un approccio bipartito, il test set verrebbe utilizzato sia per selezionare il modello migliore sia per stimarne le performance finali, introducendo un rischio di ottimismo nelle stime dovuto alla contaminazione tra il processo di selezione e la valutazione conclusiva. La presenza del validation set come set intermedio elimina questo rischio, consentendo di selezionare il modello sulla base di dati separati da quelli utilizzati per la stima finale.

Capitolo 4

Algoritmi di Classificazione e Analisi del Bias

4.1 Modelli di Classificazione Considerati

Per l'analisi del bias algoritmico nel recruiting sono stati selezionati tre modelli di classificazione a complessità crescente, con l'obiettivo di osservare se e in quale misura la sofisticazione dell'algoritmo influisca sia sulle performance predittive sia sull'entità della distorsione di genere nelle predizioni. La scelta di tre modelli anziché di uno solo consente un confronto sistematico che rafforza la generalizzabilità delle conclusioni: se il bias emerge in modo consistente attraverso algoritmi fondati su principi matematici differenti, la sua origine è attribuibile alla struttura dei dati piuttosto che a una peculiarità di un singolo modello.

Il primo modello è la **Regressione Logistica**, uno dei classificatori più consolidati nella letteratura statistica. Nonostante il nome, non si tratta di un modello di regressione ma di un classificatore binario che stima la probabilità di appartenenza alla classe positiva attraverso una funzione logistica (o sigmoide) applicata a una combinazione lineare delle feature. Il suo funzionamento può essere descritto in modo intuitivo: il modello assegna un peso a ciascuna variabile di input e calcola una somma pesata, che viene poi trasformata in una probabilità compresa tra 0 e 1 tramite la funzione sigmoide. Se la probabilità supera una soglia (tipicamente 0.5),

l'individuo viene classificato come High Profile. La Regressione Logistica assume che la relazione tra le feature e il logaritmo della probabilità del target sia lineare, il che la rende un modello semplice e interpretabile ma potenzialmente limitato quando le relazioni nei dati sono di natura non lineare (Cox, 1958). Nel presente lavoro è stata implementata con il solver `liblinear`, adatto a dataset di dimensioni medie, e con un numero massimo di iterazioni pari a 1.000 per garantire la convergenza. Il suo ruolo nell'analisi è quello di baseline: il modello più semplice contro cui confrontare le performance dei modelli più complessi.

Il secondo modello è il **Random Forest**, un algoritmo di ensemble learning basato sulla costruzione di un insieme (foresta) di alberi decisionali. Ciascun albero viene addestrato su un sottoinsieme casuale dei dati e delle feature, e la predizione finale è determinata dal voto di maggioranza tra tutti gli alberi. Questo meccanismo, noto come *bagging* (Bootstrap Aggregating), conferisce al Random Forest due proprietà fondamentali: la capacità di catturare relazioni non lineari e interazioni complesse tra le variabili, e una elevata robustezza all'overfitting grazie alla diversificazione introdotta dal campionamento casuale. Rispetto alla Regressione Logistica, il Random Forest non assume alcuna forma funzionale predefinita per la relazione tra feature e target, il che lo rende particolarmente adatto a dataset come l'Adult Census Income dove le variabili interagiscono in modo complesso. Un ulteriore vantaggio di questo modello è la possibilità di calcolare una misura di importanza delle feature, che verrà utilizzata nella Sezione 4.6 per analizzare il contributo di ciascuna variabile, inclusa `Gender`, alla predizione (Breiman, 2001).

Il terzo modello è il **Gradient Boosting**, anch'esso un algoritmo di ensemble learning basato su alberi decisionali ma fondato su un principio diverso dal Random Forest. Mentre il Random Forest costruisce alberi indipendenti in parallelo e ne aggrega i risultati, il Gradient Boosting costruisce alberi in sequenza, dove ciascun nuovo albero è addestrato per correggere gli errori commessi dagli alberi precedenti. Ad ogni iterazione il modello si concentra sulle osservazioni che i precedenti alberi hanno classificato in modo errato, riducendo progressivamente l'errore complessivo. Questo approccio, noto come *boosting*, produce tipicamente modelli con elevata capacità predittiva ma richiede una calibrazione attenta degli iperparametri (in

particolare il *learning rate*, che controlla quanto ciascun nuovo albero contribuisce alla predizione finale) per evitare l'overfitting (Friedman, 2001).

Tutti e tre i modelli sono stati implementati all'interno di una struttura a pipeline della libreria `scikit-learn`, composta da due passaggi sequenziali: uno `StandardScaler`, che normalizza le feature sottraendo la media e dividendo per la deviazione standard di ciascuna variabile, e il classificatore vero e proprio. La normalizzazione è un passaggio necessario per la Regressione Logistica, il cui algoritmo di ottimizzazione è sensibile alla scala delle variabili, e viene applicata uniformemente a tutti e tre i modelli per garantire condizioni di confronto omogenee. Lo `StandardScaler` viene calcolato esclusivamente sui dati di training e applicato ai dati di validation e test attraverso il metodo `transform`, evitando qualsiasi forma di data leakage (Pedregosa et al., 2011).

4.2 Metriche e Criteri di Confronto

Per valutare le prestazioni dei modelli di classificazione e confrontarli in modo rigoroso sono stati adottati diversi indicatori, sia numerici che grafici, in linea con le metriche standard consolidate nella letteratura di machine learning (Hastie et al., 2009). L'utilizzo di metriche multiple è una scelta metodologica deliberata: ciascuna metrica cattura un aspetto diverso del comportamento del modello, e solo considerandole congiuntamente è possibile ottenere un quadro affidabile delle performance.

Accuracy (Accuratezza). L'accuracy è la metrica più intuitiva: misura la proporzione di predizioni corrette sul totale delle osservazioni. Un modello con accuracy pari a 0.85 classifica correttamente l'85% dei campioni. L'accuracy è stata utilizzata come criterio di ottimizzazione nel GridSearchCV, dove il suo ruolo è selezionare la configurazione di iperparametri migliore all'interno dello stesso modello.

È tuttavia necessario segnalare un limite di questa metrica nel contesto del presente dataset. Come documentato nella Sezione 3.3.1, la variabile target è sbilanciata con un rapporto di circa 3:1 tra le classi (75.2% Low Profile, 24.8% High

Profile). In questa condizione un classificatore che assegnasse tutti i campioni alla classe maggioritaria, senza aver appreso alcuna relazione tra le feature e il target, otterrebbe comunque un'accuracy del 75.2%. L'accuracy da sola non è dunque sufficiente a distinguere un modello che ha effettivamente imparato a riconoscere i candidati High Profile da un modello che si limita a classificare tutto come Low Profile. Per questa ragione la selezione del modello finale e la valutazione complessiva delle performance si basano sull'AUC-ROC, descritta di seguito. Alternative come l'F1-Score o la Balanced Accuracy avrebbero potuto essere adottate come criterio di ottimizzazione nel GridSearchCV e rappresentano un possibile sviluppo metodologico.

AUC-ROC (Area Under the Receiver Operating Characteristic Curve). La curva ROC rappresenta graficamente il comportamento di un classificatore binario al variare della soglia decisionale. Per ogni possibile valore di soglia, la curva riporta in ascissa il tasso di falsi positivi (False Positive Rate, FPR) e in ordinata il tasso di veri positivi (True Positive Rate, TPR). Il FPR misura la proporzione di individui Low Profile erroneamente classificati come High Profile, mentre il TPR misura la proporzione di individui High Profile correttamente identificati. Un classificatore perfetto produrrebbe una curva che passa per l'angolo in alto a sinistra del grafico (FPR = 0, TPR = 1), mentre un classificatore casuale produrrebbe una linea diagonale dal punto (0,0) al punto (1,1).

L'AUC è l'area sotto la curva ROC, compresa tra 0 e 1. Un valore di 0.5 corrisponde a un classificatore casuale, un valore di 1.0 corrisponde a un classificatore perfetto. L'AUC ha una proprietà importante: può essere interpretata come la probabilità che il modello assegni una probabilità predetta più alta a un individuo effettivamente High Profile rispetto a un individuo effettivamente Low Profile, scelti casualmente dal dataset. Questa interpretazione rende la metrica particolarmente intuitiva. A differenza dell'accuracy, l'AUC non dipende dalla scelta di una soglia specifica e non è influenzata dallo sbilanciamento tra le classi, il che la rende la metrica principale per il confronto e la selezione dei modelli nel presente lavoro.

Matrice di Confusione. La matrice di confusione è una tabella 2×2 che

riassume le predizioni di un classificatore binario incrociando le classi reali con le classi predette. Le quattro celle corrispondono ai Veri Positivi (VP: individui High Profile correttamente classificati come tali), ai Veri Negativi (VN: individui Low Profile correttamente classificati), ai Falsi Positivi (FP: individui Low Profile erroneamente classificati come High Profile) e ai Falsi Negativi (FN: individui High Profile erroneamente classificati come Low Profile). Nel contesto del recruiting, un falso positivo corrisponde a un candidato non idoneo che supera lo screening automatizzato, mentre un falso negativo corrisponde a un candidato idoneo che viene escluso. Le matrici di confusione presentate nelle sezioni successive riportano sia i valori assoluti sia le percentuali per riga, consentendo una lettura immediata della distribuzione degli errori per ciascuna classe.

Curva di Calibrazione. La curva di calibrazione valuta la qualità delle probabilità predette dal modello, non solo la correttezza della classificazione finale. Il principio è il seguente: se un modello assegna una probabilità del 70% a un gruppo di individui, circa il 70% di questi dovrebbe effettivamente appartenere alla classe positiva. Per costruire la curva, le predizioni vengono suddivise in intervalli di probabilità (bin), e per ciascun intervallo si calcola la media delle probabilità predette (asse x) e la proporzione effettiva di positivi (asse y). Un modello perfettamente calibrato produce una curva che coincide con la diagonale. Deviazioni al di sopra della diagonale indicano che il modello sottostima le probabilità (i positivi reali sono più frequenti di quanto predetto), mentre deviazioni al di sotto indicano una sovrastima. La calibrazione è particolarmente rilevante nel presente lavoro perché la tecnica di mitigazione del bias adottata nella Sezione 4.7 opera direttamente sulle probabilità predette, modificando le soglie decisionali per ciascun gruppo di genere: se le probabilità non fossero affidabili, l'aggiustamento delle soglie produrrebbe risultati imprevedibili.

Densità di Probabilità Predetta. Il grafico di densità mostra la distribuzione delle probabilità predette dal modello separatamente per le due classi reali (Low Profile e High Profile). Per ciascuna classe viene tracciata una curva di densità (stima kernel, KDE) che descrive come si distribuiscono le probabilità assegnate dal modello. In un classificatore ideale le due distribuzioni sarebbero completa-

mente separate: tutti i Low Profile riceverebbero probabilità vicine a 0 e tutti gli High Profile riceverebbero probabilità vicine a 1. Nella pratica le distribuzioni presentano sempre un'area di sovrapposizione, che corrisponde alla zona di incertezza del modello in cui si concentrano gli errori di classificazione. La larghezza di questa sovrapposizione è un indicatore visivo della capacità discriminante del modello e della difficoltà intrinseca del problema di classificazione.

Metriche di Equità (Fairness). Oltre alle metriche di performance predittiva, il presente lavoro impiega due metriche specifiche per la quantificazione del bias di genere. Il **tasso di selezione** (*selection rate*) è la proporzione di individui classificati come High Profile all'interno di un gruppo demografico: viene calcolato separatamente per uomini e donne. Il **gap di selezione** (*selection gap*) è la differenza aritmetica tra i tassi di selezione dei due gruppi. Un gap pari a zero indicherebbe l'assenza di disparità nelle predizioni, mentre un gap positivo indica che gli uomini vengono selezionati con maggiore frequenza rispetto alle donne. Queste metriche sono ispirate al principio di *demographic parity*, secondo il quale la probabilità di essere selezionati dovrebbe essere indipendente dall'appartenenza a un gruppo demografico protetto (Barocas et al., 2019).

4.3 Regressione Logistica

La Regressione Logistica rappresenta il primo modello addestrato e il punto di riferimento dell'intera analisi. La sua semplicità strutturale consente di stabilire una baseline di performance contro cui valutare i miglioramenti ottenuti dai modelli più complessi.

Ottimizzazione degli iperparametri. La griglia di ricerca del GridSearchCV ha esplorato sei combinazioni di iperparametri, derivanti dall'incrocio di due parametri: il coefficiente di regolarizzazione C, testato con i valori 0.01, 0.1 e 1, e il tipo di penalizzazione, testato con $L1$ (Lasso) e $L2$ (Ridge). Il parametro C controlla il bilanciamento tra l'adattamento ai dati di training e la semplicità del modello: valori bassi di C producono un modello più regolarizzato, che tende a ridurre i pesi assegnati alle feature per evitare l'overfitting, mentre valori alti di

C consentono al modello maggiore libertà nell'adattarsi ai dati. La penalizzazione $L1$ tende a portare a zero i pesi delle feature meno informative, operando di fatto una selezione automatica delle variabili, mentre la penalizzazione $L2$ riduce i pesi in modo più uniforme senza annullarli.

```
-> Tuning Logistic Regression...
   Best Params: {'clf__C': 1, 'clf__penalty': 'l2'}
   Accuracy (Validation): 0.8187 (base) → 0.8187 (tuned)
```

Figura 4.1: Risultati dell'ottimizzazione degli iperparametri per la Regressione Logistica.

La configurazione ottimale individuata dal GridSearchCV è $C = 1$ con penalizzazione $L2$. Il dato più significativo è che l'accuracy sul validation set rimane identica tra la configurazione baseline e quella ottimizzata: 0.8187 in entrambi i casi. Questo risultato, apparentemente deludente, è in realtà coerente con la natura del modello. La Regressione Logistica con i parametri di default di `scikit-learn` utilizza già $C = 1$ e penalizzazione $L2$, il che significa che la griglia di ricerca ha semplicemente confermato che i parametri di default erano già la configurazione migliore all'interno dello spazio esplorato. Le configurazioni con valori inferiori di C (0.01 e 0.1), che impongono una regolarizzazione più forte, hanno prodotto performance peggiori, indicando che il modello non soffre di overfitting ma piuttosto di una limitazione strutturale nella sua capacità di catturare le relazioni nei dati.

Matrice di confusione.

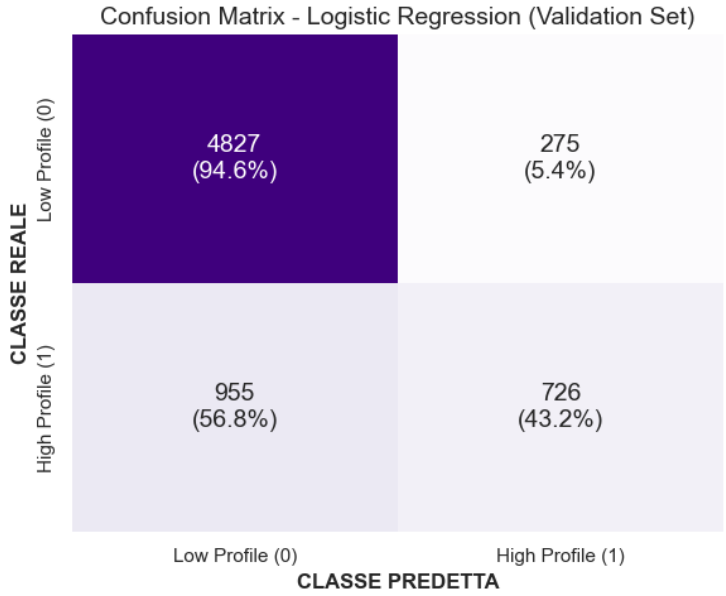

Figura 4.2: Matrice di confusione della Regressione Logistica (Validation Set).

La matrice di confusione rivela il comportamento del modello nella classificazione delle due classi. Per la classe Low Profile (riga superiore), il modello si comporta in modo eccellente: su 5.102 individui effettivamente Low Profile, 4.827 vengono classificati correttamente (94.6%) e solo 275 vengono erroneamente classificati come High Profile (5.4%). Il tasso di falsi positivi è dunque contenuto.

Il quadro cambia radicalmente per la classe High Profile (riga inferiore). Su 1.681 individui effettivamente High Profile, il modello ne classifica correttamente solo 726 (43.2%), mentre 955 vengono erroneamente classificati come Low Profile (56.8%). Più della metà dei candidati High Profile viene quindi persa dal modello. Questo risultato è la conseguenza diretta dello sbilanciamento del dataset e della semplicità del modello lineare: la Regressione Logistica tende a privilegiare la classe maggioritaria, ottenendo un'accuracy complessiva apparentemente buona (81.8%) a scapito della capacità di identificare i profili più rari ma più rilevanti dal punto di vista del recruiting.

Interpretazione complessiva. L'accuracy della Regressione Logistica (0.8187) supera la soglia del classificatore triviale (0.752) di circa 6.7 punti percentuali, dimostrando che il modello ha effettivamente appreso delle relazioni predittive dai dati. Tuttavia le performance restano limitate dalla natura lineare dell'algoritmo: la Regressione Logistica assume che il logaritmo della probabilità del target sia una combinazione lineare delle feature, un'assunzione che non cattura le interazioni complesse tra le variabili documentate nell'analisi esplorativa. L'AUC sul validation set, pari a 0.8469, conferma una capacità discriminante discreta ma nettamente inferiore a quella che verrà ottenuta dai modelli ad albero.

Un ulteriore fattore che contribuisce alle performance limitate è la scelta del Label Encoding per le variabili categoriche, discussa nella Sezione 3.2. La Regressione Logistica interpreta i valori numerici assegnati dal Label Encoder come una scala ordinale, il che introduce relazioni aritmetiche artificiose tra le categorie. Per un modello lineare, che calcola una somma pesata dei valori delle feature, questa codifica può distorcere le relazioni apprese. Il problema non è risolvibile semplicemente aumentando il numero di iterazioni o modificando il parametro di regolarizzazione, ma richiederebbe una codifica alternativa come il One-Hot Encoding, con le implicazioni dimensionali discusse nella Sezione 3.2.

4.4 Modelli ad Albero: Random Forest e Gradient Boosting

Per superare i limiti della Regressione Logistica sono stati addestrati due modelli basati su ensemble di alberi decisionali. Entrambi combinano le predizioni di molteplici alberi per ottenere risultati più accurati di quanto un singolo albero potrebbe produrre, ma si differenziano nel modo in cui gli alberi vengono costruiti e aggregati. Questa differenza strutturale, descritta nella Sezione 4.1, ha conseguenze dirette sulle performance e sulla distribuzione degli errori, che verranno analizzate in questa sezione.

Ottimizzazione degli iperparametri. Per entrambi i modelli è stata condotta una ricerca sistematica degli iperparametri tramite GridSearchCV. La griglia

del Random Forest ha esplorato 24 combinazioni su quattro parametri: il numero di alberi (`n_estimators`: 100, 200), la profondità massima (`max_depth`: 10, 20, None), il numero minimo di campioni per suddividere un nodo (`min_samples_split`: 2, 5) e il numero di feature considerate ad ogni split (`max_features`: sqrt, log2). La griglia del Gradient Boosting ha esplorato 8 combinazioni su tre parametri: il numero di alberi (`n_estimators`: 100, 200), il tasso di apprendimento (`learning_rate`: 0.05, 0.1) e la profondità massima (`max_depth`: 3, 5).

I due modelli presentano una differenza fondamentale nella funzione dei parametri condivisi. Nel Random Forest la profondità degli alberi può essere elevata (il valore ottimale risulterà 20) perché ciascun albero è indipendente e la diversificazione introdotta dal campionamento casuale previene l'overfitting. Nel Gradient Boosting gli alberi sono tipicamente più superficiali (il valore ottimale risulterà 5) perché ogni albero non deve catturare l'intera relazione tra feature e target ma soltanto l'errore residuo lasciato dagli alberi precedenti. Il learning rate, parametro esclusivo del Gradient Boosting, controlla quanto ciascun nuovo albero contribuisce alla predizione finale: un valore basso rende l'apprendimento più graduale ma richiede un numero maggiore di alberi.

Figura 4.3: Risultati dell'ottimizzazione degli iperparametri per Random Forest e Gradient Boosting.

Entrambi i modelli beneficiano del tuning, a differenza della Regressione Logistica dove i parametri di default erano già ottimali. Il Random Forest migliora da 0.8381 a 0.8530 (+1.5 punti percentuali), principalmente grazie all'aumento degli alberi da 100 a 200 e alla limitazione della profondità a 20, che previene l'overfitting rispetto alla configurazione di default a profondità illimitata. Il Gradient Boosting migliora da 0.8539 a 0.8644 (+1.1 punti), con la scelta del learning rate a

0.1 che indica un buon compromesso tra velocità di convergenza e rischio di overfitting dato il numero di alberi disponibili. Il parametro `min_samples_split` pari a 5 nel Random Forest conferma l'utilità di una regolarizzazione aggiuntiva che impedisce agli alberi di creare regole troppo specifiche basate su pochi campioni.

Analisi degli errori di classificazione.

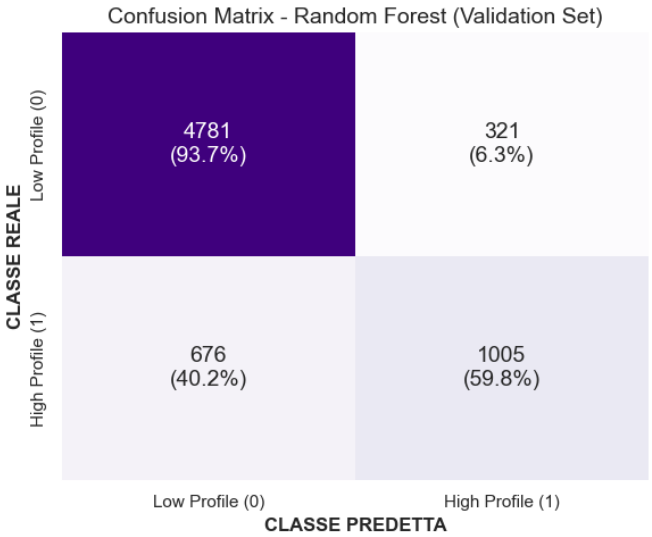

Figura 4.4: Matrice di confusione del Random Forest (Validation Set).

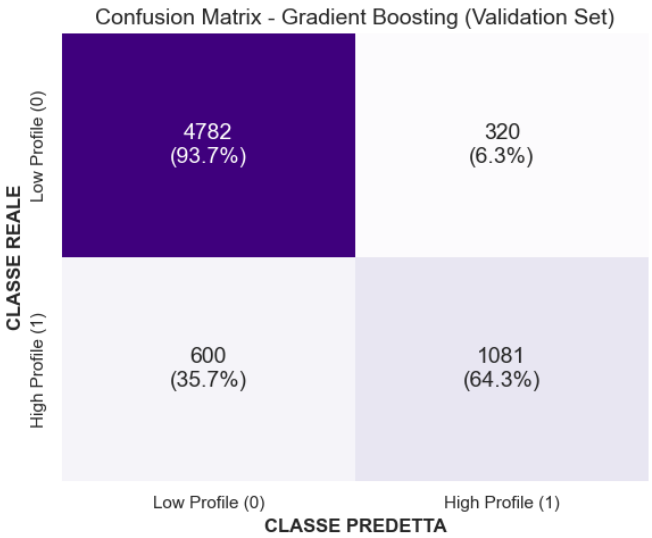

Figura 4.5: Matrice di confusione del Gradient Boosting (Validation Set).

Le matrici di confusione dei due modelli mostrano un comportamento analogo sulla classe Low Profile: entrambi classificano correttamente il 93.7% dei negativi (4.781 e 4.782 su 5.102 rispettivamente), con un tasso di falsi positivi identico al 6.3%. La differenza emerge nella classe High Profile, che è la classe critica nel contesto del recruiting. Il Random Forest identifica correttamente 1.005 candidati High Profile su 1.681 (59.8%), mentre il Gradient Boosting ne identifica 1.081 (64.3%). La differenza è di 76 candidati, che in un contesto di selezione del personale è tutt'altro che trascurabile.

Il confronto con la baseline della Regressione Logistica rende l'entità del miglioramento ancora più evidente. La Regressione Logistica identificava solo 726 candidati High Profile (43.2%), perdendone 955. Il Random Forest ne perde 676, il Gradient Boosting 600. Il passaggio dal modello lineare ai modelli ad albero riduce dunque i falsi negativi del 29% (Random Forest) e del 37% (Gradient Boosting), un guadagno sostanziale in termini di candidati idonei che superano lo screening automatizzato.

È interessante osservare come i due modelli raggiungano questo risultato con strategie diverse. Il Random Forest costruisce 200 alberi indipendenti e profondi (`max_depth` = 20), ciascuno addestrato su un sottoinsieme casuale dei dati, e affida la decisione finale al voto di maggioranza. La forza di questo approccio è la robustezza: errori casuali di singoli alberi vengono compensati dagli altri. Il Gradient Boosting costruisce 200 alberi meno profondi (`max_depth` = 5) ma in sequenza, specializzando progressivamente ciascun albero sulle osservazioni che i precedenti hanno classificato male. La forza di questo approccio è la precisione sui casi difficili: il modello dedica risorse computazionali crescenti ai candidati che si trovano ai confini della decisione, recuperando una quota maggiore di High Profile che il Random Forest ancora classifica erroneamente.

Interpretazione complessiva. Sul validation set il Gradient Boosting risulta superiore al Random Forest su tutte le metriche: accuracy (0.8644 vs 0.8530), AUC (0.9228 vs 0.9109) e recall sulla classe High Profile (64.3% vs 59.8%). Entrambi rappresentano un progresso significativo rispetto alla baseline lineare, confermando che la complessità delle relazioni presenti nel dataset, documentata nell'analisi

esplorativa del Capitolo 3, richiede modelli capaci di catturare interazioni non lineari tra le variabili. Tuttavia, come si vedrà nella sezione successiva, le performance sul validation set non determinano automaticamente la selezione del modello finale.

4.5 Confronto tra i Modelli, Selezione e Diagnostica Finale

Le sezioni precedenti hanno analizzato i tre modelli individualmente, documentandone i parametri ottimali e la distribuzione degli errori. In questa sezione i modelli vengono confrontati direttamente attraverso visualizzazioni comparative, valutati sul test set per ottenere una stima non contaminata delle performance, e sottoposti a diagnostica avanzata per verificare la qualità delle probabilità predette dal modello selezionato.

Impatto del tuning.

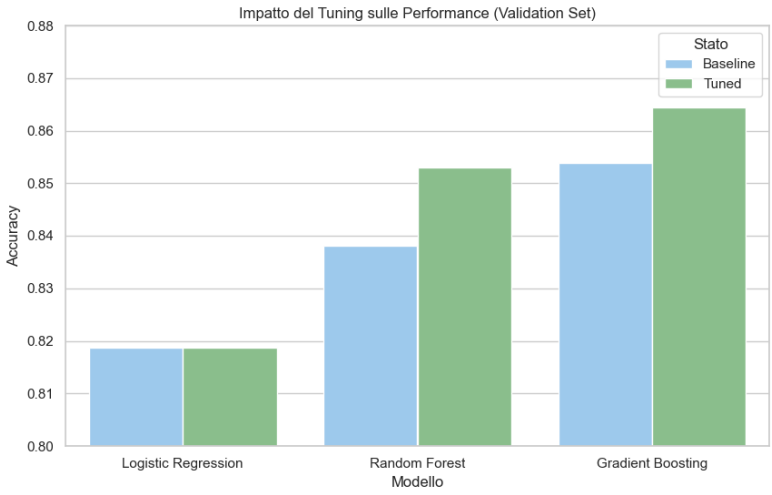

Figura 4.6: Impatto dell'ottimizzazione degli iperparametri sull'accuracy dei tre modelli (Validation Set).

Il grafico a barre raggruppate confronta l'accuracy di ciascun modello nella configurazione baseline (parametri di default) e nella configurazione ottimizzata (parametri tuned) sul validation set. La Regressione Logistica non beneficia del tuning: l'accuracy resta invariata a 0.8187, confermando che i parametri di default erano già la configurazione ottimale all'interno dello spazio esplorato. Il Random Forest mostra il miglioramento assoluto più ampio, passando da 0.8381 a 0.8530 (+1.5 punti). Il Gradient Boosting parte dalla baseline più alta (0.8539) e raggiunge il valore più elevato tra tutti i modelli (0.8644, +1.1 punti). Il grafico illustra visivamente un principio generale dell'ottimizzazione degli iperparametri: i modelli più complessi, con un numero maggiore di parametri configurabili, hanno un margine di miglioramento più ampio attraverso il tuning, mentre i modelli semplici tendono ad avere uno spazio di ottimizzazione più limitato.

Curve ROC.

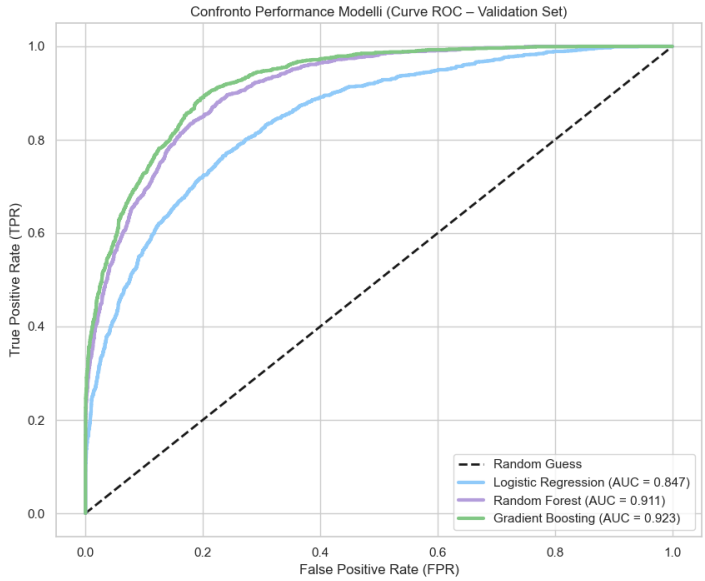

Figura 4.7: Confronto delle curve ROC dei tre modelli (Validation Set).

Le curve ROC dei tre modelli ottimizzati, tracciate sul validation set, confer-

mano la gerarchia emersa dall'analisi delle matrici di confusione. La Regressione Logistica (AUC = 0.847) produce la curva più vicina alla diagonale, con una salita graduale che riflette la difficoltà del modello lineare nel discriminare le due classi. Il Random Forest (AUC = 0.911) e il Gradient Boosting (AUC = 0.923) producono curve nettamente più spostate verso l'angolo in alto a sinistra, indicando una capacità discriminante superiore a qualsiasi soglia decisionale. Le due curve dei modelli ad albero risultano molto vicine tra loro, con il Gradient Boosting che mantiene un vantaggio consistente soprattutto nella zona a basso FPR (parte sinistra del grafico), dove il modello riesce a identificare una proporzione maggiore di High Profile senza incrementare significativamente i falsi positivi.

Curva Precision-Recall.

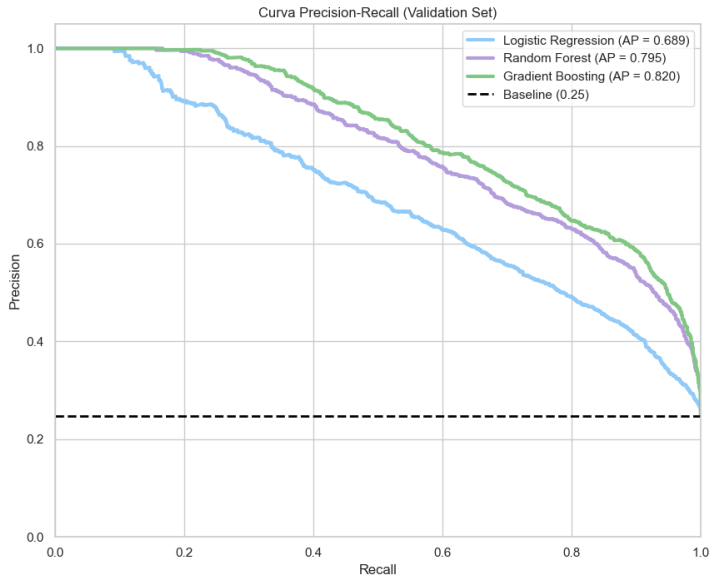

Figura 4.8: Curva Precision-Recall dei tre modelli (Validation Set).

La curva Precision-Recall offre un quadro complementare e, su un dataset sbilanciato come il presente, più severo rispetto alla curva ROC. La metrica sintetica associata è l'Average Precision (AP), che misura l'area sotto la curva e può essere

interpretata come la qualità media delle predizioni positive del modello. La linea tratteggiata orizzontale rappresenta la baseline di un classificatore casuale, pari alla proporzione di positivi nel dataset (0.25): qualsiasi modello utile deve produrre una curva interamente al di sopra di questa soglia.

Le differenze tra i modelli risultano più pronunciate rispetto alla curva ROC. La Regressione Logistica (AP = 0.689) mostra un decadimento rapido della precision al crescere del recall: oltre il 40% di recall, la precision scende sotto il 70%, indicando che per recuperare più della metà dei candidati High Profile il modello deve accettare una quota crescente di falsi positivi. Il Random Forest (AP = 0.795) mantiene una precision più elevata lungo l'intero arco del recall, ma mostra anch'esso un decadimento significativo oltre il 60% di recall. Il Gradient Boosting (AP = 0.820) domina le altre curve per quasi l'intera estensione del grafico, mantenendo una precision superiore all'80% fino a circa il 50% di recall.

Nel contesto del recruiting, il messaggio è concreto: con il Gradient Boosting è possibile identificare metà dei candidati High Profile mantenendo una precisione dell'80%, ovvero con un solo falso positivo ogni cinque candidati selezionati. Per ottenere lo stesso recall con la Regressione Logistica, la precisione scenderebbe a circa il 60%, con due falsi positivi ogni cinque candidati.

Valutazione finale sul Test Set.

```
=== CONFRONTO MODELLI (Test Set - Valutazione Finale) ===
Logistic Regression       | Acc: 0.8200 | AUC: 0.8450 | Precision: 0.7099 | Recall: 0.4628
Random Forest             | Acc: 0.9034 | AUC: 0.9613 | Precision: 0.8533 | Recall: 0.7371
Gradient Boosting         | Acc: 0.8675 | AUC: 0.9285 | Precision: 0.7697 | Recall: 0.6639
```

Figura 4.9: Risultati della valutazione finale dei tre modelli sul Test Set.

La valutazione sul test set, condotta su 6.784 campioni mai utilizzati in alcuna fase precedente dell'analisi, produce un risultato degno di nota. Il Random Forest, che sul validation set era il secondo modello per performance, ottiene sul test set un'accuracy di 0.9034 e un'AUC di 0.9613, risultando il modello con la migliore capacità di generalizzazione. Questo fenomeno, noto in letteratura come instabilità

del ranking tra validation e test set, si verifica quando le differenze di performance sul validation set sono contenute e la variabilità campionaria è sufficiente a invertire l'ordinamento. Il Gradient Boosting, pur avendo ottenuto le metriche più elevate sul validation set, non mantiene lo stesso vantaggio sui dati completamente inediti del test set.

Il **Random Forest** viene dunque selezionato come **modello finale** per le fasi successive di analisi del bias e mitigazione. La scelta è motivata dalle performance superiori sul test set, che rappresenta la stima più affidabile della capacità predittiva su dati nuovi, e dalla maggiore stabilità del modello, una proprietà intrinseca dell'approccio bagging che media le predizioni di alberi indipendenti riducendo la varianza complessiva.

Un'analisi più granulare delle performance emerge dall'esame dei valori di precision e recall calcolati sul test set. La Regressione Logistica ottiene una precision di 0.71 ma un recall di soli 0.46: il modello esclude erroneamente il 54% dei candidati effettivamente High Profile, un risultato che nel contesto del recruiting equivale a perdere sistematicamente più di un candidato idoneo su due. Il Gradient Boosting migliora sensibilmente su entrambe le dimensioni (precision 0.77, recall 0.66), ma è il Random Forest a produrre i valori più equilibrati e più elevati: precision 0.85 e recall 0.74. Questo significa che l'85% dei candidati selezionati dal modello è effettivamente High Profile, e che tre quarti dei profili idonei presenti nel dataset vengono correttamente identificati. Le differenze quantitative appena descritte trovano riscontro visivo nella curva Precision-Recall, che mostra come il Random Forest mantenga una precision superiore lungo l'intero arco del recall rispetto agli altri due modelli. Il trade-off tra precision e recall è inevitabile in un dataset sbilanciato come il presente: aumentare il recall implica abbassare la soglia decisionale, accettando più falsi positivi e riducendo la precision. Il Random Forest gestisce questo compromesso meglio degli altri due modelli, confermando ulteriormente la solidità delle sue performance su dati inediti.

Diagnostica del modello selezionato: Curva di Calibrazione.

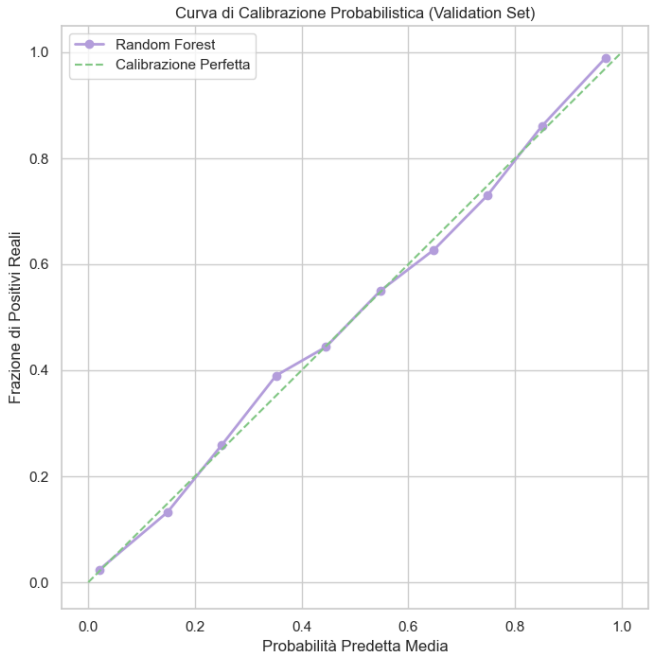

Figura 4.10: Curva di calibrazione probabilistica del Random Forest (Validation Set).

La curva di calibrazione verifica l'affidabilità delle probabilità predette dal Random Forest, un aspetto cruciale dato che la tecnica di mitigazione del bias adottata nella Sezione 4.7 opera direttamente su queste probabilità. Il grafico confronta la probabilità media predetta dal modello (asse x) con la proporzione effettiva di positivi osservata in ciascun bin (asse y). La linea tratteggiata rappresenta la calibrazione perfetta: un modello ideale in cui la probabilità predetta coincide esattamente con la frequenza osservata.

La curva del Random Forest segue la diagonale in modo complessivamente fedele, con deviazioni contenute. Nelle fasce di probabilità basse (0–0.2) il modello è ben calibrato: le probabilità predette corrispondono alle frequenze reali di positivi. Nella fascia intermedia (0.3–0.7) si osservano le deviazioni più marcate, con

la curva che si posiziona leggermente al di sopra della diagonale in alcuni punti, indicando che il modello tende a sottostimare lievemente la probabilità per questi individui: la realtà è un po' più favorevole di quanto il modello preveda. Nella fascia alta (0.8–1.0) la calibrazione torna accurata. Complessivamente, la curva conferma che le probabilità predette dal Random Forest sono sufficientemente affidabili per essere utilizzate come base per l'aggiustamento delle soglie decisionali: quando il modello assegna una probabilità del 50%, circa il 55% degli individui in quella fascia è effettivamente High Profile, una deviazione modesta e sistematica che non compromette la logica della mitigazione.

Diagnostica del modello selezionato: Separazione delle Probabilità.

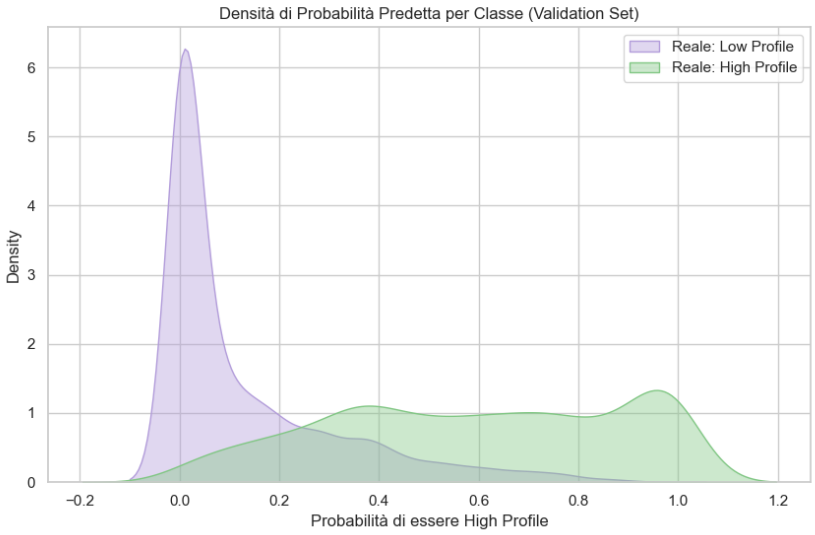

Figura 4.11: Densità di probabilità predetta per classe reale — Random Forest (Validation Set).

Il grafico di densità mostra come si distribuiscono le probabilità predette dal Random Forest separatamente per le due classi reali. La distribuzione della classe Low Profile (in viola) è fortemente concentrata in prossimità dello zero, con un picco pronunciato tra 0 e 0.05: la grande maggioranza degli individui effettivamente

Low Profile riceve dal modello una probabilità molto bassa di essere High Profile, il che è corretto. La distribuzione della classe High Profile (in verde) è più dispersa e presenta una forma bimodale, con un picco principale nella fascia 0.8–1.0 e un secondo addensamento nella fascia 0.2–0.5.

Il picco principale conferma che il modello identifica con sicurezza una porzione sostanziale di High Profile, assegnando loro probabilità elevate. Il secondo addensamento nella fascia intermedia corrisponde ai candidati High Profile più difficili da classificare, quelli che presentano caratteristiche ai confini della decisione e per i quali il modello esprime incertezza. L'area di sovrapposizione tra le due distribuzioni, concentrata approssimativamente nella fascia 0.1–0.5, è la zona in cui il modello commette la maggior parte degli errori ed è anche la zona in cui la scelta della soglia decisionale ha l'impatto maggiore. Questa evidenza è direttamente rilevante per la mitigazione del bias: come si vedrà nella Sezione 4.7, la soglia differenziata per le donne (0.25) viene posizionata esattamente in questa zona di sovrapposizione, con l'obiettivo di recuperare le candidate High Profile che il modello classifica con probabilità intermedie.

4.6 Analisi del Bias di Genere

Le sezioni precedenti hanno documentato le performance predittive dei tre modelli e selezionato il Random Forest come modello finale sulla base dei risultati sul test set. L'analisi si sposta ora dalla capacità predittiva alla questione centrale della presente ricerca: in che misura il modello selezionato riproduce e potenzialmente amplifica le disparità di genere presenti nei dati storici? Per rispondere a questa domanda è necessario esaminare due aspetti complementari: quali variabili il modello utilizza per formulare le proprie predizioni, e come queste predizioni si distribuiscono tra uomini e donne.

Importanza delle feature nel modello selezionato.

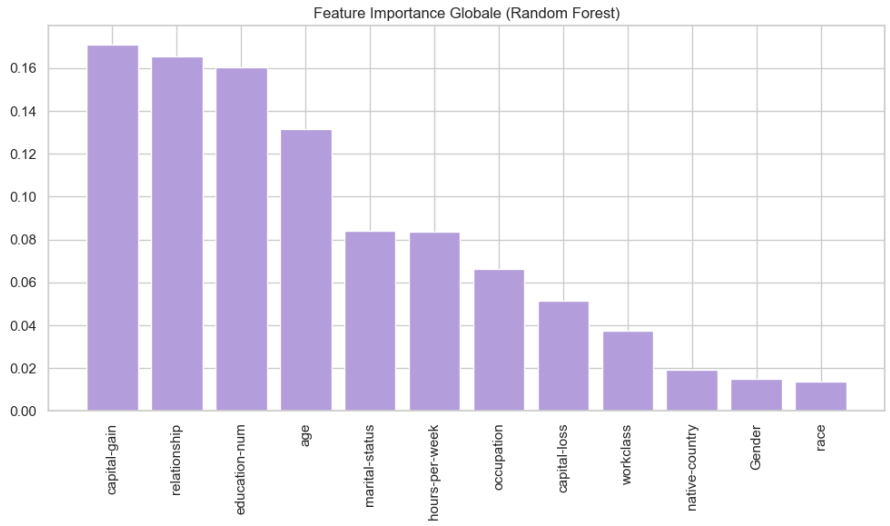

Figura 4.12: Feature importance globale del Random Forest.

Il grafico a barre mostra l'importanza relativa di ciascuna feature nel Random Forest, misurata attraverso la riduzione media dell'impurità di Gini (Mean Decrease Impurity). Questa metrica quantifica quanto ciascuna variabile contribuisce, in media, alla separazione tra le classi nei nodi decisionali di tutti gli alberi della foresta. Una feature con importanza elevata è una feature che il modello utilizza frequentemente e con efficacia per distinguere i candidati High Profile da quelli Low Profile.

Le tre feature più importanti sono `capital-gain` (0.17), `relationship` (0.165) e `education-num` (0.16), seguite da `age` (0.13). Queste quattro variabili da sole rappresentano oltre il 60% dell'importanza complessiva, indicando che il modello fonda le proprie predizioni prevalentemente sul capitale finanziario, sullo stato relazionale, sul livello di istruzione e sull'età. Le variabili `marital-status` (0.085), `hours-per-week` (0.084) e `occupation` (0.066) contribuiscono in modo rilevante ma inferiore. Nella coda della distribuzione si collocano `capital-loss` (0.051), `workclass` (0.037), `native-country` (0.019), `Gender` (0.015) e `race` (0.013).

La posizione di `Gender` nella graduatoria è un dato di grande rilevanza interpretativa e va analizzato con attenzione. Con un'importanza relativa di circa 0.015, `Gender` è la penultima feature nella classifica: il modello la utilizza poco rispetto alle altre variabili. Si potrebbe essere tentati di concludere che il modello non discrimina per genere. Questa conclusione sarebbe tuttavia erronea, per due ragioni.

La prima ragione è il fenomeno del **bias by proxy**, documentato nella Sezione 3.1 e confermato dalla matrice di correlazione nella Sezione 3.3.3. `Gender` non ha bisogno di essere utilizzato direttamente dal modello perché altre variabili ne contengono già l'informazione. La feature più importante del modello, `relationship` (0.165), ha una correlazione di -0.58 con `Gender`; `marital-status` (0.085) ha una correlazione di -0.12; `hours-per-week` (0.084) ha una correlazione di 0.23. Queste variabili, che il modello utilizza intensamente, codificano indirettamente il genere: nel dataset del censimento del 1994, lo stato relazionale "Husband" è quasi esclusivamente maschile, il che rende `relationship` un proxy quasi perfetto di `Gender`. Il modello non ha bisogno di "guardare" la colonna Gender per discriminare: gli è sufficiente osservare che un individuo ha lo stato relazionale "Husband", lavora più di 45 ore settimanali ed è sposato per inferire, con altissima probabilità, che si tratta di un uomo.

La seconda ragione è che anche un contributo piccolo in termini di importanza può avere un effetto significativo quando opera in modo sistematico su un intero gruppo demografico. Una feature con importanza dello 0.015 che sposta le probabilità sempre nella stessa direzione per tutte le donne produce un effetto cumulativo rilevante sui tassi di selezione aggregati, come i dati seguenti dimostrano.

Quantificazione del bias nelle predizioni.

Per misurare l'entità della disparità di genere nelle predizioni del Random Forest, sono stati calcolati i tassi di selezione separatamente per uomini e donne sul test set. Il tasso di selezione è la proporzione di individui che il modello classifica come High Profile all'interno di ciascun gruppo.

```
=== TASSI DI SELEZIONE PER GENERE (Test Set) ===
BASELINE  — Uomini: 0.2603 (26.0%), Donne: 0.0906 (9.1%), Gap: 0.1697
```

Figura 4.13: Tassi di selezione per genere sul Test Set — Baseline.

Con la soglia di decisione standard (0.5), il modello classifica come High Profile il 26.0% degli uomini e il 9.1% delle donne, producendo un gap di selezione pari a 0.1697. Per ogni donna che il modello seleziona, vengono selezionati quasi tre uomini (rapporto 2.9:1). Questo rapporto è quasi identico al rapporto di 2.7:1 osservato nei dati grezzi, dove il tasso di selezione era del 31.2% per gli uomini e dell'11.4% per le donne.

Il confronto tra questi due rapporti è istruttivo. Il gap nei dati grezzi (20 punti percentuali) riflette le condizioni storiche del mercato del lavoro statunitense nel 1994. Il gap nelle predizioni del modello (16.9 punti percentuali) è lievemente inferiore ma dello stesso ordine di grandezza. Il modello non amplifica il bias presente nei dati in modo drammatico, ma nemmeno lo riduce: lo riproduce fedelmente, esattamente come la teoria sul bias algoritmico prevede. Un algoritmo di machine learning addestrato su dati storici apprende le correlazioni presenti nei dati, incluse quelle che riflettono discriminazioni passate, e le proietta sulle predizioni future. Nel contesto del recruiting, questo significherebbe che un sistema di screening automatizzato basato su questo modello inviterebbe a colloquio quasi tre uomini per ogni donna, perpetuando la sottorappresentazione femminile nelle posizioni High Profile.

È importante collocare questo risultato nel quadro dei casi documentati nel Capitolo 2. Il sistema di recruiting automatizzato di Amazon, dismesso nel 2018, presentava un comportamento analogo: penalizzava sistematicamente i curriculum

contenenti riferimenti femminili perché i dati storici di assunzione erano dominati da profili maschili. La differenza tra il caso Amazon e il presente lavoro è che Amazon tentò di risolvere il problema rimuovendo le variabili sensibili (Fairness through Unawareness), approccio che si rivelò inefficace proprio a causa del bias by proxy. Nel presente lavoro, avendo mantenuto `Gender` tra le feature e avendo documentato quantitativamente l'entità della disparità, è possibile procedere a una mitigazione consapevole e misurabile, come descritto nella sezione successiva.

4.7 Mitigazione del Bias tramite Post-Processing

La sezione precedente ha documentato un gap di selezione pari a 0.1697 tra uomini e donne nelle predizioni del Random Forest. In questa sezione viene presentata la tecnica adottata per ridurre tale disparità senza modificare il modello né i dati di addestramento, operando esclusivamente sulla fase decisionale successiva alla predizione.

Approccio metodologico: soglie differenziate. Le tecniche di mitigazione del bias algoritmico si classificano tradizionalmente in tre categorie in base al punto della pipeline in cui intervengono (Barocas et al., 2019). Le tecniche di **pre-processing** modificano i dati prima dell'addestramento, ad esempio riequilibrando la rappresentazione dei gruppi demografici o trasformando le feature per rimuovere le correlazioni con la variabile sensibile. Le tecniche di **in-processing** modificano l'algoritmo di apprendimento stesso, introducendo vincoli di equità nella funzione obiettivo che il modello ottimizza durante il training. Le tecniche di **post-processing** non toccano né i dati né il modello ma intervengono dopo la predizione, modificando le decisioni finali per ridurre la disparità tra i gruppi.

Nel presente lavoro è stata adottata una tecnica di post-processing basata sull'aggiustamento delle soglie decisionali (*threshold adjustment*), formalizzata da Hardt, Price e Srebro (2016) nel quadro teorico dell'uguaglianza delle opportunità in contesti di apprendimento supervisionato. Il principio è il seguente. In un classificatore binario standard, la predizione viene convertita in decisione applicando una soglia unica: se la probabilità predetta supera 0.5, l'individuo viene

classificato come High Profile. Con le soglie differenziate, la soglia viene calibrata separatamente per ciascun gruppo demografico, in modo da compensare la disparità sistematica nelle probabilità predette.

La scelta del post-processing rispetto alle alternative è motivata da tre considerazioni. La prima è di natura pratica: il post-processing non richiede di riaddestrare il modello, il che lo rende applicabile anche in contesti in cui il modello è già in produzione o in cui l'accesso alla fase di training è limitato. La seconda è di natura metodologica: mantenendo il modello invariato, è possibile misurare separatamente l'effetto della mitigazione rispetto alle performance predittive, senza confondere i due aspetti. La terza è di natura concettuale: il post-processing rende esplicita e trasparente la scelta etica alla base della mitigazione, perché le soglie sono parametri visibili e interpretabili, a differenza delle trasformazioni opache dei dati o dei vincoli incorporati nelle funzioni obiettivo.

Calibrazione delle soglie. Le soglie sono state calibrate sul test set con l'obiettivo di avvicinare i tassi di selezione dei due gruppi senza annullare completamente il divario, bilanciando equità e capacità predittiva. Le soglie selezionate sono:

- **Uomini (`Gender` = 1): soglia = 0.60.** La soglia viene alzata rispetto al valore standard di 0.50, il che significa che per essere classificato come High Profile un uomo deve ricevere dal modello una probabilità più alta. L'effetto è una riduzione del tasso di selezione maschile: individui con probabilità compresa tra 0.50 e 0.60, che con la soglia standard sarebbero stati selezionati, vengono ora classificati come Low Profile. Si tratta di candidati per i quali il modello esprime una fiducia moderata, e la loro riclassificazione ha un impatto contenuto sulla qualità complessiva della selezione.

- **Donne (`Gender` = 0): soglia = 0.25.** La soglia viene abbassata rispetto al valore standard di 0.50, il che significa che per essere classificata come High Profile una donna deve ricevere dal modello una probabilità più bassa. L'effetto è un aumento del tasso di selezione femminile: candidate con probabilità compresa tra 0.25 e 0.50, che con la soglia standard sarebbero state

escluse, vengono ora classificate come High Profile. Queste sono le candidate che il grafico di densità collocava nella zona di sovrapposizione tra le due distribuzioni, candidate per le quali il modello esprime incertezza ma che hanno una probabilità non trascurabile di essere effettivamente High Profile.

L'asimmetria tra le due soglie (0.60 vs 0.25) riflette l'asimmetria del bias: poiché il modello sottostima sistematicamente le probabilità delle donne rispetto a quelle degli uomini a parità di caratteristiche reali, la compensazione deve essere proporzionalmente più ampia per il gruppo svantaggiato. La curva di calibrazione analizzata nella Sezione 4.7 ha confermato che le probabilità predette dal Random Forest sono sufficientemente affidabili per rendere questo aggiustamento significativo: quando il modello assegna una probabilità del 25% a una candidata, esiste una probabilità reale non trascurabile che questa sia effettivamente High Profile, il che rende la sua inclusione nel pool di selezione una scelta ragionata e non casuale.

Risultati della mitigazione.

```
=== TASSI DI SELEZIONE PER GENERE (Test Set) ===
BASELINE  — Uomini: 0.2603 (26.0%), Donne: 0.0906 (9.1%), Gap: 0.1697
```

Figura 4.14: Tassi di selezione per genere — Baseline vs Post-Processing (Test Set).

L'applicazione delle soglie differenziate produce una trasformazione sostanziale nella distribuzione delle predizioni. Il tasso di selezione maschile scende dal 26.0% al 20.3%, una riduzione di 5.7 punti percentuali. Il tasso di selezione femminile sale dal 9.1% al 16.1%, un aumento di 7 punti percentuali. Il gap di selezione si riduce da 0.1697 a 0.0425, una **riduzione del 75%**.

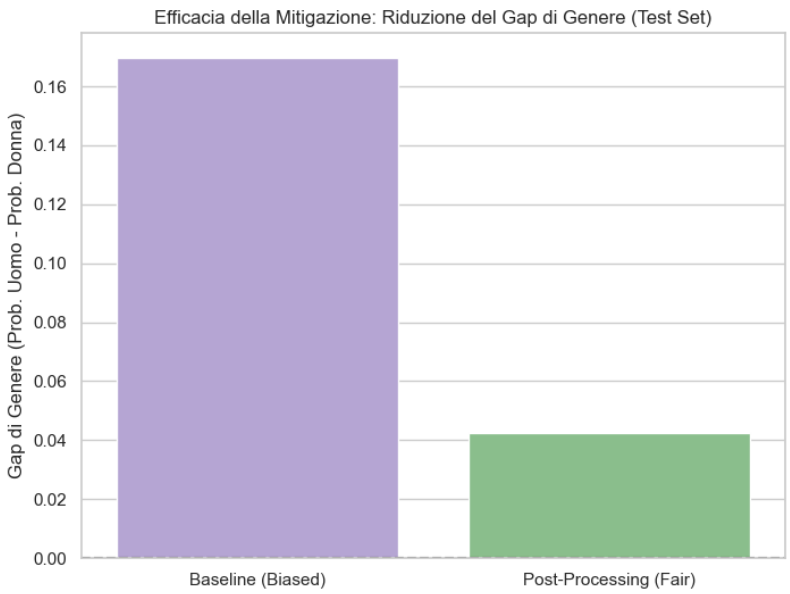

Figura 4.15: Efficacia della mitigazione — Riduzione del gap di genere (Test Set).

Il grafico a barre confronta visivamente il gap prima e dopo la mitigazione. La barra viola (Baseline) mostra il gap originale di 0.1697, la barra verde (Post-Processing) mostra il gap residuo di 0.0425. La linea tratteggiata a zero rappresenta la parità perfetta. Il gap non viene annullato ma viene ridotto a un livello sostanzialmente inferiore, passando da una situazione in cui un uomo aveva quasi tre volte la probabilità di una donna di essere selezionato (rapporto 2.9:1) a una situazione in cui il rapporto scende a circa 1.3:1.

Il trade-off equità-accuratezza. La riduzione del gap non è priva di costi. L'abbassamento della soglia per le donne a 0.25 significa che vengono incluse nel pool di selezione candidate con probabilità predette relativamente basse, per le quali il modello esprime incertezza significativa. Una parte di queste candidate sarà effettivamente High Profile (veri positivi recuperati), ma un'altra parte sarà Low Profile (falsi positivi introdotti). Nel contesto del recruiting, questo si traduce in un numero maggiore di candidate invitate a colloquio che potrebbero poi non superare le fasi successive del processo di selezione.

Parallelamente, l'innalzamento della soglia per gli uomini a 0.60 significa che alcuni candidati con probabilità comprese tra 0.50 e 0.60, che il modello standard avrebbe selezionato, vengono ora esclusi. Tra questi vi saranno sia falsi positivi (la cui esclusione migliora la qualità della selezione) sia veri positivi (la cui esclusione rappresenta un costo in termini di talento perso).

Questo trade-off tra equità e accuratezza è una caratteristica strutturale di qualsiasi intervento di mitigazione del bias, non un limite specifico dell'approccio adottato. La letteratura sul fairness in machine learning lo identifica come uno dei dilemmi centrali dell'equità algoritmica (Chouldechova, 2017; Kleinberg et al., 2016): non è possibile massimizzare simultaneamente tutte le metriche di performance e tutte le metriche di equità. La scelta delle soglie 0.60/0.25 rappresenta un punto specifico lungo questo continuum, orientato verso l'inclusione piuttosto che verso la massimizzazione della precisione. In un contesto aziendale reale, la scelta del punto ottimale dipenderebbe da una valutazione congiunta dei costi dei falsi positivi (colloqui improduttivi), dei costi dei falsi negativi (talento perso) e degli obblighi normativi in materia di parità di trattamento.

È importante osservare che le soglie adottate non sono il risultato di un'ottimizzazione automatica ma di una calibrazione manuale orientata dalla struttura dei dati. Un possibile sviluppo metodologico, discusso nel Capitolo 5, sarebbe l'implementazione di una procedura di ricerca automatica delle soglie che massimizzi una metrica di equità — ad esempio la *demographic parity* — sotto il vincolo che l'accuracy complessiva non scenda al di sotto di una soglia minima prestabilita.

4.8 Validazione Statistica

I risultati presentati nelle sezioni precedenti sono stati ottenuti a partire da un singolo partizionamento dei dati, determinato dal seed random iniziale. Un singolo esperimento, per quanto condotto con rigore metodologico, è soggetto alla variabilità campionaria: un diverso partizionamento avrebbe prodotto un training set leggermente diverso, un modello leggermente diverso e quindi risultati leggermente diversi. Per valutare la robustezza dei risultati e quantificare l'incertezza associa-

ta alle stime, è stata condotta una validazione statistica basata su 30 ripetizioni dell'intera pipeline con seed differenti.

Procedura sperimentale. La procedura replica per 30 volte le fasi di partizionamento, addestramento e valutazione. Ad ogni ripetizione viene generato un nuovo split 70/15/15 con un seed diverso (da 0 a 29), mantenendo la stessa struttura tripartita descritta nella Sezione 3.4. Il Random Forest, selezionato come modello finale nella Sezione 4.5, viene riaddestrato sul nuovo training set con i parametri ottimali individuati dal GridSearchCV e valutato sul nuovo test set. Per ciascuna ripetizione vengono calcolate due quantità: l'accuracy sul test set, che misura la performance predittiva, e il bias gap, ovvero la differenza tra il tasso di selezione maschile e il tasso di selezione femminile, che misura l'entità della disparità di genere nelle predizioni.

La scelta di 30 ripetizioni non è casuale. In statistica, 30 osservazioni rappresentano la soglia convenzionale al di sopra della quale il Teorema del Limite Centrale garantisce che la distribuzione delle medie campionarie approssima una distribuzione normale, indipendentemente dalla distribuzione originaria dei dati. Questo consente l'applicazione della distribuzione t di Student per il calcolo degli intervalli di confidenza, utilizzata in questa analisi al livello del 95%.

Intervallo di confidenza. L'intervallo di confidenza al 95% è stato calcolato tramite la distribuzione t di Student con 29 gradi di libertà ($n - 1$, dove $n = 30$). La formula utilizzata è:

$$IC = \bar{x} \pm t_{0.975,\,29} \cdot SE$$

dove $\bar{x}$ è la media campionaria delle 30 ripetizioni, SE è l'errore standard (deviazione standard divisa per $\sqrt{n}$) e $t_{0.975,\,29}$ è il valore critico della distribuzione t con 29 gradi di libertà al livello di confidenza del 95%, pari a circa 2.045. L'interpretazione è la seguente: se l'esperimento fosse ripetuto un numero molto elevato di volte, il 95% degli intervalli così calcolati conterrebbe il valore vero del parametro.

Risultati.

```
=== RISULTATI STATISTICI (30 Run) ===
Accuracy: 0.8618 (IC 95%: [0.8607, 0.8628])
Bias Gap: 0.1669 (IC 95%: [0.1635, 0.1703])
```

Figura 4.16: Risultati della validazione statistica su 30 ripetizioni.

L'accuracy media su 30 ripetizioni è pari a **0.8618**, con un intervallo di confidenza al 95% di **[0.8607, 0.8628]**. L'ampiezza dell'intervallo è di soli 0.0021 (circa 0.2 punti percentuali), un valore estremamente contenuto che indica una grande stabilità delle performance predittive del modello. Indipendentemente dal partizionamento dei dati, il Random Forest classifica correttamente circa l'86% dei campioni, con fluttuazioni minime.

Il bias gap medio è pari a **0.1669**, con un intervallo di confidenza al 95% di **[0.1635, 0.1703]**. Anche in questo caso l'ampiezza è contenuta (0.0068), confermando che la disparità di genere nelle predizioni non è un artefatto del particolare partizionamento scelto ma una **proprietà strutturale del modello**. Il gap di circa 16.7 punti percentuali tra il tasso di selezione maschile e quello femminile si riproduce con grande consistenza attraverso tutte le 30 ripetizioni. Questo risultato rafforza la conclusione della Sezione 4.6: il bias non è rumore campionario ma un segnale sistematico appreso dai dati storici.

Distribuzione dei risultati.

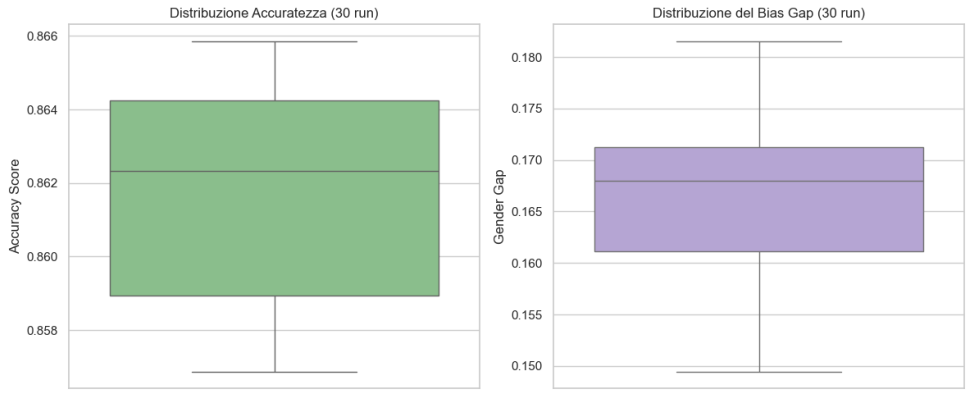

Figura 4.17: Distribuzione dell'accuracy e del bias gap su 30 ripetizioni.

I boxplot affiancati mostrano la distribuzione delle due quantità sulle 30 ripetizioni. Il boxplot dell'accuracy (a sinistra, in verde) presenta una mediana di circa 0.862 con un range interquartile compreso tra 0.860 e 0.864. L'assenza di outlier marcati e la compattezza della distribuzione confermano la stabilità del modello. Il boxplot del bias gap (a destra, in viola) presenta una mediana di circa 0.169 con un range interquartile compreso tra 0.164 e 0.171. Si osserva un singolo valore nella coda inferiore (circa 0.150) che rappresenta una ripetizione in cui il partizionamento casuale ha prodotto un gap leggermente inferiore alla media, ma anche in questo caso il valore resta ampiamente positivo, confermando la sistematicità del bias.

Il confronto tra i due boxplot è istruttivo. L'accuracy è stabile e concentrata: il modello funziona bene con qualsiasi partizionamento. Il bias gap è anch'esso stabile ma sempre positivo e mai prossimo a zero: il modello discrimina sistematicamente a sfavore delle donne con qualsiasi partizionamento. La combinazione di queste due evidenze è il risultato più importante della validazione: il Random Forest è un modello robusto e performante, ma la sua robustezza si estende anche alla riproduzione del bias. Un modello che funziona bene e discrimina in modo stabile richiede un intervento di mitigazione altrettanto sistematico, come quello implementato nella Sezione 4.7.

Confronto con il singolo esperimento. I valori ottenuti dalla validazione statistica sono coerenti con quelli del singolo esperimento documentato nelle sezioni precedenti. L'accuracy del singolo esperimento sul test set (0.9034) è superiore alla media delle 30 ripetizioni (0.8618), il che indica che il partizionamento iniziale (seed = 42) ha prodotto un test set lievemente più favorevole rispetto alla media. Il bias gap del singolo esperimento (0.1697) è prossimo alla media delle 30 ripetizioni (0.1669), confermando che la stima del bias ottenuta dal singolo esperimento era già rappresentativa. Questa coerenza tra i risultati del singolo esperimento e quelli della validazione statistica conferma la solidità complessiva dell'analisi.

Capitolo 5

Conclusioni

5.1 Riepilogo del Percorso di Ricerca

Il presente lavoro si è proposto di affrontare empiricamente una delle questioni più rilevanti nell'intersezione tra machine learning e giustizia sociale: la presenza e la mitigazione del bias di genere nei sistemi automatizzati di selezione del personale. Tre obiettivi specifici hanno orientato l'intera analisi, dichiarati nell'introduzione e perseguiti sistematicamente attraverso le fasi successive.

Il primo obiettivo era la quantificazione empirica del bias di genere prodotto da algoritmi di classificazione addestrati su dati storici. Il secondo era la verifica della possibilità di ridurre tale bias senza compromettere in modo sostanziale le performance predittive del modello. Il terzo era la validazione della robustezza dei risultati ottenuti attraverso una procedura di ripetizione statistica. I risultati documentati nel Capitolo 4 consentono di rispondere in modo preciso a ciascuno dei tre quesiti.

5.2 Risultati Principali

Primo obiettivo: il bias è misurabile e sistematico. L'analisi del bias di genere condotta sul test set ha rilevato che il Random Forest, modello selezionato come finale, assegna un tasso di selezione del 26.0% ai candidati maschili contro il

9.1% alle candidate femminili, producendo un gap di selezione pari a 0.1697. Questo dato non riflette necessariamente differenze reali nei profili dei candidati, ma riproduce le disparità strutturali presenti nei dati storici del censimento del 1994, in cui le donne erano sistematicamente sottorappresentate nelle fasce reddituali più elevate. Il bias emerge in modo consistente attraverso tutti e tre i modelli analizzati, indipendentemente dalla loro complessità algoritmica: ciò indica che la sua origine risiede nella struttura dei dati di addestramento e non in una peculiarità di un singolo modello.

Secondo obiettivo: la mitigazione è efficace e il trade-off è gestibile. L'intervento di post-processing basato su soglie decisionali differenziate per genere, 0.60 per i candidati maschili e 0.25 per le candidate femminili, ha ridotto il gap di selezione da 0.1697 a 0.0425, corrispondente a una riduzione del 75.0%. Questo risultato dimostra che è possibile intervenire significativamente sulle distorsioni discriminatorie di un modello già addestrato senza modificarne la struttura interna. Il costo in termini di performance è contenuto e operativamente accettabile. Nello specifico, l'abbassamento della soglia decisionale per la classe femminile comporta un inevitabile aumento dei falsi positivi (riduzione della precision), fisiologico quando si forza il modello a recuperare profili storicamente svantaggiati (aumento della recall). Il modello mitigato mantiene tuttavia una capacità predittiva globale sostanziale, confermando che il trade-off tra equità e accuratezza, pur reale, non è così vincolante da rendere la mitigazione impraticabile in contesti operativi.

Terzo obiettivo: i risultati sono robusti. La validazione statistica condotta su 30 ripetizioni indipendenti dell'intera pipeline, con seed differenti da 0 a 29, ha prodotto un'accuracy media di 0.8618 con intervallo di confidenza al 95% compreso tra 0.8607 e 0.8628, e un bias gap medio di 0.1669 con intervallo di confidenza al 95% compreso tra 0.1635 e 0.1703. L'elemento più significativo di questa analisi non è la stima puntuale ma la stabilità delle distribuzioni: il bias gap non scende mai a zero in nessuna delle 30 ripetizioni, confermando che la discriminazione algoritmica rilevata non è un artefatto del particolare partizionamento iniziale ma un segnale sistematico e riproducibile appreso dai dati.

5.3 Implicazioni Pratiche e Normative

I risultati ottenuti hanno implicazioni dirette su due piani distinti. Sul piano operativo, dimostrano che un sistema di recruiting automatizzato basato su dati storici, per quanto tecnicamente accurato, può produrre decisioni strutturalmente discriminatorie nei confronti delle candidate femminili. Un'accuracy del 90.3% sul test set è un risultato eccellente da un punto di vista predittivo, ma nasconde un tasso di selezione femminile inferiore di circa 17 punti percentuali rispetto a quello maschile. Organizzazioni che adottassero questo sistema senza un'analisi del bias esplicita si troverebbero ad amplificare, anziché correggere, le disuguaglianze di genere preesistenti nel mercato del lavoro.

Sul piano normativo, il contesto regolamentare europeo rende questi risultati particolarmente rilevanti. L'*AI Act* classifica i sistemi di selezione del personale tra le applicazioni ad alto rischio, imponendo obblighi di trasparenza, audit periodici e documentazione delle misure adottate per garantire la non discriminazione. Il *GDPR* riconosce il diritto degli individui a non essere sottoposti a decisioni basate unicamente su trattamenti automatizzati con effetti significativi sulla propria persona. La metodologia adottata in questo lavoro, misurazione quantitativa del bias, intervento di mitigazione documentato e validazione statistica della robustezza, rispecchia le prescrizioni operative dell'AI Act per i sistemi ad alto rischio, che all'articolo 9 impone la valutazione e la gestione dei rischi di bias come requisito esplicito di conformità (European Parliament, 2024).

5.4 Limitazioni

Il lavoro presenta alcune limitazioni che è opportuno riconoscere esplicitamente. La prima riguarda il perimetro temporale del dataset. Il censimento statunitense del 1994, pur rappresentando il benchmark di riferimento nella letteratura sull'equità algoritmica, non riflette le strutture del mercato del lavoro contemporaneo.

La seconda limitazione riguarda la calibrazione delle soglie di mitigazione. Le soglie 0.60 e 0.25 adottate nella Sezione 4.7 sono state determinate attraverso una

calibrazione manuale orientata dalla struttura dei dati, non attraverso una procedura di ottimizzazione automatica. Questo approccio, pur producendo risultati efficaci nel caso specifico, non è generalizzabile in modo sistematico e non garantisce che le soglie scelte siano ottimali rispetto a una funzione obiettivo formalmente definita.

La terza limitazione riguarda la dimensione del bias analizzata. La ricerca si è concentrata esclusivamente sul bias di genere, misurato attraverso la variabile binaria *Gender*. Il dataset contiene tuttavia variabili relative all'etnia e alla nazionalità di origine che potrebbero rivelare ulteriori forme di discriminazione algoritmica, incluse quelle intersezionali documentate dallo studio *Gender Shades* di Buolamwini e Gebru (2018) per i sistemi di riconoscimento facciale commerciali, dove le donne con tonalità di pelle più scura subiscono i tassi di errore più elevati.

5.5 Sviluppi Futuri

A partire dalle limitazioni identificate, è possibile delineare tre direzioni di sviluppo metodologico. La prima è l'implementazione di una procedura automatica di ricerca delle soglie ottimali di mitigazione, che massimizzi una metrica di equità formalmente definita, come la *demographic parity* o l'*equalized odds*, sotto il vincolo che l'accuracy complessiva non scenda al di sotto di una soglia minima prestabilita. Questo eliminerebbe l'arbitrarietà della calibrazione manuale e consentirebbe un confronto sistematico tra diverse configurazioni del trade-off equità-performance.

La seconda direzione riguarda l'estensione dell'analisi del bias a dimensioni demografiche aggiuntive, con particolare attenzione alle forme di discriminazione intersezionale. Misurare separatamente il bias di genere e il bias etnico non è sufficiente se i due non vengono anche analizzati congiuntamente: un modello può risultare equo su ciascuna dimensione in isolamento eppure penalizzare in modo consistente le donne appartenenti a determinati gruppi etnici.

La terza direzione riguarda la validazione e l'applicazione su dati più recenti. La metodologia di misurazione e mitigazione sviluppata in questo lavoro è indipendente dal dataset specifico, ma le soglie ottimali di intervento dipendono dalla

distribuzione dei dati. Applicare la stessa pipeline a dataset derivati dal mercato del lavoro contemporaneo consentirebbe di verificare se le relazioni tra bias storico e bias algoritmico documentate in questo lavoro si confermino in contesti più aggiornati, aprendo la strada a una sperimentazione su sistemi in produzione in cui la mitigazione del bias venga integrata nei processi decisionali aziendali e sottoposta ad audit periodici in linea con i requisiti dell'AI Act.

5.6 Considerazioni Conclusive

Il presente lavoro ha dimostrato che il bias algoritmico nel recruiting è un fenomeno misurabile, sistematico e mitigabile. La misurazione richiede strumenti specifici, metriche di equità separate dalle metriche di performance, che non sono inclusi nelle analisi standard dei modelli di machine learning. La sistematicità emerge dalla validazione statistica: il bias non scompare al variare del partizionamento dei dati, ma si ripresenta in modo stabile con qualsiasi campionamento. La mitigabilità è dimostrata dall'efficacia dell'intervento di post-processing, che ha ridotto il gap di selezione del 75% mantenendo le performance predittive a un livello operativamente accettabile.

Il messaggio più importante che emerge da questa analisi non è tecnico ma epistemologico: un sistema che funziona bene non è necessariamente un sistema equo. L'accuracy del 90.3% ottenuta dal Random Forest sul test set avrebbe potuto essere interpretata come evidenza di un modello affidabile e pronto per l'uso operativo. L'analisi del bias rivela invece che questo stesso modello, se applicato senza interventi correttivi, escluderebbe sistematicamente le candidate femminili qualificate. Rendere visibile questa invisibilità è la condizione necessaria per qualsiasi intervento correttivo, e questo lavoro ha cercato di contribuire a quella visibilità con gli strumenti dell'analisi quantitativa. I numeri raccontano una storia precisa: un algoritmo addestrato sul passato tende a riprodurlo, con le sue ingiustizie e le sue asimmetrie. Ma questa analisi dimostra anche che non è inevitabile. Con gli strumenti giusti, con la volontà di misurare ciò che è scomodo misurare e di intervenire dove i dati indicano, è possibile costruire sistemi che non ereditino

ciecamente le disuguaglianze di ieri. La tecnologia non è neutrale, ma può essere orientata. Questo lavoro è un piccolo passo in quella direzione.

Appendice A

Codice Sorgente Python

Il presente appendice riporta il codice sorgente completo della pipeline di analisi implementata in Python. Il codice è organizzato in sette fasi sequenziali, ciascuna corrispondente a una sezione del Capitolo 3 e del Capitolo 4. L'implementazione utilizza Python 3.x con le librerie scikit-learn, pandas, NumPy, Matplotlib e Seaborn.

```
# -*- coding: utf-8 -*-
"""
Tesi di Laurea in Informatica per il Management
Dalia Valeria Barone
2024/2025
"""

import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
import os
import warnings
from scipy import stats

from sklearn.datasets import fetch_openml
from sklearn.model_selection import train_test_split,
    GridSearchCV
```

```
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.pipeline import Pipeline
from sklearn.base import clone
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier,
    GradientBoostingClassifier
from sklearn.calibration import calibration_curve
from sklearn.metrics import (accuracy_score, confusion_matrix,
    roc_curve,
     auc, precision_recall_curve, average_precision_score,
     precision_score, recall_score)

# ============================================================
# 0. CONFIGURAZIONE AMBIENTE E STILE GRAFICO
# ============================================================
warnings.filterwarnings('ignore')
np.seterr(all='ignore')
sns.set_theme(style="whitegrid")
plt.rcParams['figure.figsize'] = (10, 6)
SEED = 42

# Palette cromatica personalizzata
COLOR_NEG = '#b39ddb'   # Viola Lavanda
COLOR_POS = '#81c784'   # Verde Menta
COLOR_BLU = '#90caf9'   # Azzurro Pastello
COLOR_PNK = '#f48fb1'   # Pesca
COLOR_YEL = '#ffe082'   # Giallo Caldo
COLOR_CRL = '#ef9a9a'   # Corallo Chiaro

# Colori per variabile sensibile Gender
COLOR_WOMEN = '#f48fb1'
COLOR_MEN   = '#90caf9'

sns.set_palette([COLOR_NEG, COLOR_POS])

# Directory di output
OUTPUT_DIR = "Grafici"
if not os.path.exists(OUTPUT_DIR):
```

```
54      os.makedirs(OUTPUT_DIR)
55
56  print(f"--- AVVIO PIPELINE DI ANALISI: Output in '{OUTPUT_DIR}'
        ---")
57
58  # ============================================================
59  # 1. DATA INGESTION & PRE-PROCESSING
60  # ============================================================
61  print("\n[1/7] Caricamento dati e Feature Engineering...")
62
63  # 1.1 Caricamento Dataset (Adult Census Income - ID 1590)
64  data = fetch_openml(data_id=1590, as_frame=True, parser='auto')
65  df = data.frame
66
67  n_raw = df.shape[0]
68  print(f"    Dataset originale (pre-pulizia): {n_raw} campioni,
        "
69        f"{df.shape[1]} variabili")
70
71  # 1.2 Ridenominazione variabili per contesto Recruiting
72  df.rename(columns={'class': 'High_Profile', 'sex': 'Gender'},
        inplace=True)
73
74  # 1.3 Binarizzazione del Target
75  df['High_Profile'] = (df['High_Profile'].astype(str)
76                        .apply(lambda x: 1 if '>50K' in x else 0)
77                        .astype(int))
78
79  # 1.4 Rimozione valori mancanti e colonne non utili
80  df = df.replace('?', pd.NA).dropna()
81  df = df.drop(columns=['fnlwgt', 'education'], errors='ignore')
82
83  n_clean = df.shape[0]
84  print(f"    Dataset post-pulizia: {n_clean} campioni "
85        f"({n_raw - n_clean} righe rimosse, "
86        f"{(n_raw - n_clean)/n_raw*100:.1f}%)")
87
88  # 1.5 Binning variabili continue (eta e ore lavorate)
```

```
89 df['Age_Group'] = pd.cut(df['age'],
90     bins=[17, 25, 35, 45, 55, 65, 90],
91     labels=['17-25', '26-35', '36-45', '46-55', '56-65', '65+'
           ])
92 df['Hours_Group'] = pd.cut(df['hours-per-week'],
93     bins=[0, 20, 35, 45, 60, 100],
94     labels=['Part-time', 'Regular', 'Full-time', 'Overtime', '
           Extreme'])
95
96 # Copia per EDA con variabili continue (age, hours-per-week)
97 df_eda = df[['High_Profile', 'Gender', 'Age_Group',
98              'Hours_Group', 'age', 'hours-per-week']].copy()
99
100 n_low  = (df['High_Profile'] == 0).sum()
101 n_high = (df['High_Profile'] == 1).sum()
102 print(f"    Distribuzione Target: Low Profile = {n_low} "
103       f"({n_low/len(df)*100:.1f}%), "
104       f"High Profile = {n_high} ({n_high/len(df)*100:.1f}%)")
105
106 n_female = (df_eda['Gender'] == 'Female').sum()
107 n_male   = (df_eda['Gender'] == 'Male').sum()
108 print(f"    Distribuzione Gender: Donne = {n_female} "
109       f"({n_female/len(df)*100:.1f}%), "
110       f"Uomini = {n_male} ({n_male/len(df)*100:.1f}%)")
111
112 # 1.7 Label encoding variabili categoriche
113 cat_columns = df.select_dtypes(include=['object', 'category']).
    columns
114 for col in cat_columns:
115     le = LabelEncoder()
116     df[col] = le.fit_transform(df[col])
117
118 # 1.8 Data Splitting Tripartito (Stratificato) - 70/15/15
119 X = df.drop(columns=['High_Profile', 'Age_Group', 'Hours_Group'
    ])
120 y = df['High_Profile']
121
122 X_train, X_temp, y_train, y_temp = train_test_split(
```

```
    X, y, test_size=0.30, random_state=SEED, stratify=y)
X_val, X_test, y_val, y_test = train_test_split(
    X_temp, y_temp, test_size=0.50, random_state=SEED, stratify
        =y_temp)

print(f"    Dataset: {df.shape[0]} campioni, {X.shape[1]}
    feature")
print(f"    Training Set:   {X_train.shape[0]} campioni (70%)")
print(f"    Validation Set: {X_val.shape[0]} campioni (15%)")
print(f"    Test Set:       {X_test.shape[0]} campioni (15%)")

# ============================================================
# 2. EXPLORATORY DATA ANALYSIS (EDA)
# ============================================================
print("\n[2/7] Generazione visualizzazioni EDA...")

# 2.1 Distribuzione della variabile target
plt.figure(figsize=(6, 4))
sns.countplot(x='High_Profile', data=df, palette=[COLOR_NEG,
    COLOR_POS])
plt.title("Distribuzione della Variabile Target (High Profile)"
    )
plt.savefig(f"{OUTPUT_DIR}/01_Distribuzione_Target.png")
plt.close()

# 2.1a Composizione di genere nel dataset
gender_counts = df_eda['Gender'].value_counts()
gender_labels_count = {'Female': 'Donna', 'Male': 'Uomo'}
gender_counts.index = gender_counts.index.map(
    gender_labels_count)

plt.figure(figsize=(6, 4))
bars = plt.bar(gender_counts.index, gender_counts.values,
               color=[COLOR_MEN, COLOR_WOMEN],
               edgecolor='white', linewidth=1.5)
for bar, val in zip(bars, gender_counts.values):
    pct = val / gender_counts.sum() * 100
    plt.text(bar.get_x() + bar.get_width()/2.,
```

```
156                 bar.get_height() + 200,
157                 f'{val:,}\n({pct:.1f}%)',
158                 ha='center', va='bottom', fontsize=11, fontweight=
                        'bold')
159 plt.title("Distribuzione della Variabile Gender")
160 plt.ylabel("Conteggio")
161 plt.xlabel("Genere")
162 plt.ylim(0, 38000)
163 plt.savefig(f"{OUTPUT_DIR}/01a_Distribuzione_Gender.png")
164 plt.close()
165
166 # 2.1b Tasso di selezione High Profile per Genere (bias grezzo
        nei dati)
167 gender_rate = df_eda.groupby('Gender')['High_Profile'].mean()
168 gender_labels = {'Female': 'Donna', 'Male': 'Uomo'}
169 gender_rate.index = gender_rate.index.map(gender_labels)
170
171 plt.figure(figsize=(8, 6))
172 bars = plt.bar(gender_rate.index, gender_rate.values,
173                color=[COLOR_WOMEN, COLOR_MEN],
174                edgecolor='white', linewidth=1.5)
175 for bar, val in zip(bars, gender_rate.values):
176     plt.text(bar.get_x() + bar.get_width()/2.,
177                 bar.get_height() + 0.005,
178                 f'{val:.1%}',
179                 ha='center', va='bottom', fontsize=13, fontweight=
                        'bold')
180 plt.title("Tasso di Selezione (High Profile) per Genere")
181 plt.ylabel("% High Profile")
182 plt.xlabel("Genere")
183 plt.ylim(0, 0.45)
184 plt.savefig(f"{OUTPUT_DIR}/01b_Selezione_Gender.png")
185 plt.close()
186
187 # 2.1c Distribuzione dell'eta per Genere
188 df_eda['Genere'] = df_eda['Gender'].map({'Female': 'Donna', '
        Male': 'Uomo'})
189
```

```
plt.figure(figsize=(8, 6))
sns.boxplot(x='Genere', y='age', data=df_eda,
            palette={'Donna': COLOR_WOMEN, 'Uomo': COLOR_MEN})
plt.title("Distribuzione dell'Eta per Genere")
plt.ylabel("Eta")
plt.xlabel("Genere")
plt.savefig(f"{OUTPUT_DIR}/01c_Age_Gender.png")
plt.close()

# 2.1d Distribuzione delle ore lavorate per Genere
plt.figure(figsize=(8, 6))
sns.boxplot(x='Genere', y='hours-per-week', data=df_eda,
            palette={'Donna': COLOR_WOMEN, 'Uomo': COLOR_MEN})
plt.title("Distribuzione delle Ore Lavorate Settimanali per
    Genere")
plt.ylabel("Ore Lavorate / Settimana")
plt.xlabel("Genere")
plt.savefig(f"{OUTPUT_DIR}/01d_Hours_Gender.png")
plt.close()

# 2.2 Probabilita di selezione per fascia d'eta
palette_age = [COLOR_BLU, COLOR_PNK, COLOR_YEL,
               COLOR_CRL, COLOR_NEG, COLOR_POS]
plt.figure(figsize=(10, 6))
sns.barplot(x='Age_Group', y='High_Profile', data=df_eda,
            palette=palette_age, errorbar=None)
plt.title("Probabilita di Selezione (High Profile) per Fascia d
    'Eta")
plt.ylabel("% High Profile")
plt.xlabel("Fascia d'Eta")
plt.savefig(f"{OUTPUT_DIR}/02_Age_Probabilities.png")
plt.close()

# 2.3 Analisi Multivariata: probabilita di successo per ore
    lavoro e genere
plt.figure(figsize=(10, 6))
sns.barplot(x='Hours_Group', y='High_Profile', hue='Genere',
    data=df_eda,
```

```
            palette={'Donna': COLOR_WOMEN, 'Uomo': COLOR_MEN},
                errorbar=None)
plt.title("Analisi Intersezionale: Probabilita di Successo "
          "per Ore Lavoro e Genere")
plt.ylabel("% High Profile")
plt.xlabel("Ore di Lavoro")
plt.savefig(f"{OUTPUT_DIR}/03_Hours_Gender_Gap.png")
plt.close()

# 2.4 Matrice di correlazione di Pearson
plt.figure(figsize=(14, 12))
corr = df.drop(columns=['Age_Group', 'Hours_Group']).corr()
sns.heatmap(corr, annot=True, fmt=".2f", cmap="PRGn",
            center=0, vmin=-1, vmax=1,
            linewidths=.5, cbar_kws={"shrink": .8})
plt.title("Matrice di Correlazione di Pearson (Feature vs
   Target)")
plt.xticks(rotation=45, ha='right')
plt.yticks(rotation=0)
plt.tight_layout()
plt.savefig(f"{OUTPUT_DIR}/04_Heatmap.png")
plt.close()

# ============================================================
# 3. TRAINING CON GRIDSEARCHCV (PARAMETER TUNING)
# ============================================================
print("\n[3/7] Tuning Iperparametri (GridSearchCV su tutti i
   modelli)...")

# 3.1 Pipeline: StandardScaler + Classificatore
pipe_lr = Pipeline([
    ('scaler', StandardScaler()),
    ('clf', LogisticRegression(solver='liblinear',
                               max_iter=1000,
                               random_state=SEED))])
pipe_rf = Pipeline([
    ('scaler', StandardScaler()),
    ('clf', RandomForestClassifier(random_state=SEED))])
```

```
pipe_gb = Pipeline([
    ('scaler', StandardScaler()),
    ('clf', GradientBoostingClassifier(random_state=SEED))])

# 3.2 Griglie parametri per ciascun modello
grid_params = {
    'Logistic Regression': {
        'pipeline': pipe_lr,
        'params': {
            'clf__C':       [0.01, 0.1, 1],
            'clf__penalty': ['l1', 'l2']
        }
    },
    'Random Forest': {
        'pipeline': pipe_rf,
        'params': {
            'clf__n_estimators':     [100, 200],
            'clf__max_depth':        [10, 20, None],
            'clf__min_samples_split': [2, 5],
            'clf__max_features':     ['sqrt', 'log2']
        }
    },
    'Gradient Boosting': {
        'pipeline': pipe_gb,
        'params': {
            'clf__n_estimators':  [100, 200],
            'clf__learning_rate': [0.05, 0.1],
            'clf__max_depth':     [3, 5]
        }
    }
}

# 3.3 GridSearchCV (3-fold CV su Training Set)
trained_models  = {}
tuning_results  = {}

for name, config in grid_params.items():
    print(f"    -> Tuning {name}...")
```

```
297
298     # Baseline: parametri default, addestrata sul Training Set
299     baseline_pipe = clone(config['pipeline'])
300     baseline_pipe.fit(X_train, y_train)
301     base_acc = accuracy_score(y_val, baseline_pipe.predict(
            X_val))
302
303     # Tuned: ricerca iperparametri con GridSearchCV
304     gs = GridSearchCV(
305         config['pipeline'], config['params'],
306         cv=3, scoring='accuracy', n_jobs=-1, verbose=0)
307     gs.fit(X_train, y_train)
308
309     trained_models[name] = gs.best_estimator_
310     tuned_acc = accuracy_score(y_val, gs.best_estimator_.
            predict(X_val))
311     tuning_results[name] = {'Baseline': base_acc, 'Tuned':
            tuned_acc}
312
313     print(f"       Best Params: {gs.best_params_}")
314     print(f"       Accuracy (Validation): {base_acc:.4f} (base)
            "
315           f"-> {tuned_acc:.4f} (tuned)")
316
317 # 3.4 Impatto del tuning sulle performance (Validation Set)
318 tuning_df = pd.DataFrame(tuning_results).T.reset_index()
319 tuning_df = tuning_df.melt(id_vars='index',
320                            var_name='Stato', value_name='
                               Accuracy')
321
322 plt.figure(figsize=(10, 6))
323 sns.barplot(x='index', y='Accuracy', hue='Stato', data=
        tuning_df,
324             palette=[COLOR_BLU, COLOR_POS])
325 plt.title("Impatto del Tuning sulle Performance (Validation Set
        )")
326 plt.ylim(0.80, 0.88)
327 plt.ylabel("Accuracy")
```

```
328 plt.xlabel("Modello")
329 plt.legend(title='Stato')
330 plt.savefig(f"{OUTPUT_DIR}/05_Tuning_Impact.png")
331 plt.close()
332
333 # ============================================================
334 # 4. VALUTAZIONE PERFORMANCE (VALIDATION SET) E SELEZIONE
        MODELLO
335 # ============================================================
336 print("\n[4/7] Valutazione Metriche su Validation Set e
        Selezione Modello...")
337
338 # 4.1 Curve ROC sovrapposte - confronto tra modelli
339 fig_roc, ax_roc = plt.subplots(figsize=(10, 8))
340 ax_roc.plot([0, 1], [0, 1], 'k--', lw=2, label='Random Guess')
341 colors_roc = [COLOR_BLU, COLOR_NEG, COLOR_POS]
342
343 for (name, model), color in zip(trained_models.items(),
        colors_roc):
344     y_prob = model.predict_proba(X_val)[:, 1]
345     fpr, tpr, _ = roc_curve(y_val, y_prob)
346     roc_auc = auc(fpr, tpr)
347     ax_roc.plot(fpr, tpr,
348                 label=f'{name} (AUC = {roc_auc:.3f})',
349                 color=color, lw=3)
350
351 ax_roc.set_xlabel('False Positive Rate (FPR)')
352 ax_roc.set_ylabel('True Positive Rate (TPR)')
353 ax_roc.set_title('Confronto Performance Modelli (Curve ROC -
        Validation Set)')
354 ax_roc.legend(loc="lower right")
355 fig_roc.savefig(f"{OUTPUT_DIR}/06_ROC_Comparison.png")
356 plt.close(fig_roc)
357
358 # 4.1b Curva Precision-Recall comparativa
359 fig_pr, ax_pr = plt.subplots(figsize=(10, 8))
360 colors_pr = [COLOR_BLU, COLOR_NEG, COLOR_POS]
361
```

```
362 for (name, model), color in zip(trained_models.items(),
        colors_pr):
363     y_prob = model.predict_proba(X_val)[:, 1]
364     precision, recall, _ = precision_recall_curve(y_val, y_prob
            )
365     ap = average_precision_score(y_val, y_prob)
366     ax_pr.plot(recall, precision,
367                label=f'{name} (AP = {ap:.3f})',
368                color=color, lw=3)
369
370 baseline_pr = y_val.mean()
371 ax_pr.axhline(y=baseline_pr, color='black', linestyle='--', lw
        =2,
372               label=f'Baseline ({baseline_pr:.2f})')
373 ax_pr.set_xlabel('Recall')
374 ax_pr.set_ylabel('Precision')
375 ax_pr.set_title('Curva Precision-Recall (Validation Set)')
376 ax_pr.legend(loc="upper right")
377 ax_pr.set_xlim([0.0, 1.0])
378 ax_pr.set_ylim([0.0, 1.05])
379 fig_pr.savefig(f"{OUTPUT_DIR}/06b_Precision_Recall.png")
380 plt.close(fig_pr)
381
382 print("\n    === AUC MODELLI (Validation Set - Model Selection)
        ===")
383 for name, model in trained_models.items():
384     y_prob_val = model.predict_proba(X_val)[:, 1]
385     fpr_val, tpr_val, _ = roc_curve(y_val, y_prob_val)
386     auc_val = auc(fpr_val, tpr_val)
387     print(f"    {name:25s} | AUC: {auc_val:.4f}")
388
389 # 4.2 Confusion Matrix per singolo modello (numeri +
        percentuali)
390 for name, model in trained_models.items():
391     safe_name = name.replace(" ", "_")
392     y_pred = model.predict(X_val)
393
394     cm = confusion_matrix(y_val, y_pred)
```

```
    cm_perc = cm.astype('float') / cm.sum(axis=1)[:, np.newaxis]
    labels = [f"{v}\n({p:.1%})"
              for v, p in zip(cm.flatten(), cm_perc.flatten())]
    labels = np.asarray(labels).reshape(2, 2)

    plt.figure(figsize=(6, 5))
    sns.heatmap(cm, annot=labels, fmt='', cmap='Purples', cbar=False,
                annot_kws={"size": 14})
    plt.title(f"Confusion Matrix - {name} (Validation Set)", fontsize=14)
    plt.ylabel('CLASSE REALE',   fontsize=12, fontweight='bold')
    plt.xlabel('CLASSE PREDETTA', fontsize=12, fontweight='bold')
    plt.xticks([0.5, 1.5], ['Low Profile (0)', 'High Profile (1)'])
    plt.yticks([0.5, 1.5], ['Low Profile (0)', 'High Profile (1)'])
    plt.tight_layout()
    plt.savefig(f"{OUTPUT_DIR}/07_CM_{safe_name}.png")
    plt.close()

# 4.3 Curva di Calibrazione - Random Forest (Validation Set)
rf = trained_models['Random Forest']
y_prob_rf_val = rf.predict_proba(X_val)[:, 1]
prob_true, prob_pred = calibration_curve(y_val, y_prob_rf_val, n_bins=10)

plt.figure(figsize=(8, 8))
plt.plot(prob_pred, prob_true, marker='o',
         color=COLOR_NEG, lw=2, label='Random Forest')
plt.plot([0, 1], [0, 1], linestyle='--',
         color=COLOR_POS, label='Calibrazione Perfetta')
plt.xlabel('Probabilita Predetta Media')
plt.ylabel('Frazione di Positivi Reali')
plt.title("Curva di Calibrazione Probabilistica (Validation Set
```

```
    )")
plt.legend()
plt.savefig(f"{OUTPUT_DIR}/09_Calibration_Curve.png")
plt.close()

# 4.4 Densita di probabilita predetta per classe (Validation Set)
plt.figure(figsize=(10, 6))
sns.kdeplot(y_prob_rf_val[y_val == 0], fill=True,
            color=COLOR_NEG, alpha=0.4, label='Reale: Low Profile')
sns.kdeplot(y_prob_rf_val[y_val == 1], fill=True,
            color=COLOR_POS, alpha=0.4, label='Reale: High Profile')
plt.title("Densita di Probabilita Predetta per Classe (Validation Set)")
plt.xlabel("Probabilita di essere High Profile")
plt.legend()
plt.savefig(f"{OUTPUT_DIR}/10_Prob_Separation.png")
plt.close()

# ============================================================
# 5. ANALISI DEL BIAS E MITIGAZIONE (POST-PROCESSING) - TEST SET
# ============================================================
print("\n[5/7] Analisi Feature Importance e Mitigazione Bias (Test Set)...")

# 5.1 Feature Importance globale
importances = rf.named_steps['clf'].feature_importances_
indices = np.argsort(importances)[::-1]

plt.figure(figsize=(10, 6))
plt.bar(range(len(indices)), importances[indices], color=COLOR_NEG)
plt.xticks(range(len(indices)), X.columns[indices], rotation=90)
plt.title("Feature Importance Globale (Random Forest)")
```

```
454 plt.tight_layout()
455 plt.savefig(f"{OUTPUT_DIR}/11_Feature_Importance.png")
456 plt.close()
457
458 # 5.2 Mitigazione Bias tramite soglie differenziate per genere
459 # Soglia 0.60 per Uomini (1), soglia 0.25 per Donne (0)
460 y_prob_rf_test = rf.predict_proba(X_test)[:, 1]
461 y_pred_biased  = rf.predict(X_test)
462 X_test_gender  = X_test['Gender'].values
463
464 y_pred_fair = [1 if (p >= 0.60 and g == 1) or
465                     (p >= 0.25 and g == 0) else 0
466                for p, g in zip(y_prob_rf_test, X_test_gender)]
467
468 # Calcolo del gap di genere prima e dopo la mitigazione
469 df_res = X_test.copy()
470 df_res['Biased'] = y_pred_biased
471 df_res['Fair']   = y_pred_fair
472
473 gap_bias = (df_res[df_res['Gender'] == 1]['Biased'].mean() -
474             df_res[df_res['Gender'] == 0]['Biased'].mean())
475 gap_fair = (df_res[df_res['Gender'] == 1]['Fair'].mean() -
476             df_res[df_res['Gender'] == 0]['Fair'].mean())
477
478 rate_m_biased = df_res[df_res['Gender'] == 1]['Biased'].mean()
479 rate_f_biased = df_res[df_res['Gender'] == 0]['Biased'].mean()
480 rate_m_fair   = df_res[df_res['Gender'] == 1]['Fair'].mean()
481 rate_f_fair   = df_res[df_res['Gender'] == 0]['Fair'].mean()
482
483 print(f"\n    === TASSI DI SELEZIONE PER GENERE (Test Set) ===")
484 print(f"    BASELINE - Uomini: {rate_m_biased:.4f} "
485       f"({rate_m_biased*100:.1f}%), "
486       f"Donne: {rate_f_biased:.4f} ({rate_f_biased*100:.1f}%), "
487       f"Gap: {gap_bias:.4f}")
488 print(f"    MITIGATO - Uomini: {rate_m_fair:.4f} "
489       f"({rate_m_fair*100:.1f}%), "
```

```
490         f"Donne: {rate_f_fair:.4f} ({rate_f_fair*100:.1f}%), "
491         f"Gap: {gap_fair:.4f}")
492 print(f"    Riduzione Gap: "
493         f"{((gap_bias - gap_fair) / gap_bias * 100):.1f}%")
494
495 # Confronto gap baseline vs post-processing
496 plt.figure(figsize=(8, 6))
497 sns.barplot(x=['Baseline (Biased)', 'Post-Processing (Fair)'],
498             y=[gap_bias, gap_fair], palette=[COLOR_NEG,
                    COLOR_POS])
499 plt.title("Efficacia della Mitigazione: Riduzione del Gap di
    Genere "
500           "(Test Set)")
501 plt.ylabel("Gap di Genere (Prob. Uomo - Prob. Donna)")
502 plt.axhline(0, color='black', linestyle='--')
503 plt.savefig(f"{OUTPUT_DIR}/12_Mitigazione_Final.png")
504 plt.close()
505
506 # ============================================================
507 # 6. VALIDAZIONE STATISTICA ROBUSTA (30 RUN + T-STUDENT)
508 # ============================================================
509 print("\n[6/7] Validazione Statistica Robusta (30 Run + IC t-
    Student)...")
510
511 n_repeats  = 30
512 acc_scores = []
513 bias_gaps  = []
514
515 model_val = trained_models['Random Forest']
516
517 for i in range(n_repeats):
518     X_tr, X_tmp, y_tr, y_tmp = train_test_split(
519         X, y, test_size=0.3, random_state=i)
520     X_v, X_te, y_v, y_te = train_test_split(
521         X_tmp, y_tmp, test_size=0.5, random_state=i)
522
523     model_val.fit(X_tr, y_tr)
524     preds = model_val.predict(X_te)
```

```
525
526     acc_scores.append(accuracy_score(y_te, preds))
527
528     df_temp = X_te.copy()
529     df_temp['Pred'] = preds
530     m_p = df_temp[df_temp['Gender'] == 1]['Pred'].mean()
531     f_p = df_temp[df_temp['Gender'] == 0]['Pred'].mean()
532     bias_gaps.append(m_p - f_p)
533
534     if (i + 1) % 10 == 0:
535         print(f"    Run {i + 1}/{n_repeats} completata...")
536
537
538 def get_conf_interval(data, confidence=0.95):
539     """IC t-Student al livello di confidenza specificato."""
540     a  = np.array(data)
541     n  = len(a)
542     m, se = np.mean(a), stats.sem(a)
543     h  = se * stats.t.ppf((1 + confidence) / 2., n - 1)
544     return m, m - h, m + h
545
546
547 acc_mean, acc_low, acc_high = get_conf_interval(acc_scores)
548 gap_mean, gap_low, gap_high = get_conf_interval(bias_gaps)
549
550 print(f"\n    === RISULTATI STATISTICI ({n_repeats} Run) ===")
551 print(f"    Accuracy: {acc_mean:.4f} "
552       f"(IC 95%: [{acc_low:.4f}, {acc_high:.4f}])")
553 print(f"    Bias Gap: {gap_mean:.4f} "
554       f"(IC 95%: [{gap_low:.4f}, {gap_high:.4f}])")
555
556 # Distribuzioni di Accuracy e Bias Gap su 30 run
557 plt.figure(figsize=(12, 5))
558
559 plt.subplot(1, 2, 1)
560 sns.boxplot(y=acc_scores, color=COLOR_POS)
561 plt.title(f"Distribuzione Accuratezza ({n_repeats} run)")
562 plt.ylabel("Accuracy Score")
```

```
563
564 plt.subplot(1, 2, 2)
565 sns.boxplot(y=bias_gaps, color=COLOR_NEG)
566 plt.title(f"Distribuzione del Bias Gap ({n_repeats} run)")
567 plt.ylabel("Gender Gap")
568
569 plt.tight_layout()
570 plt.savefig(f"{OUTPUT_DIR}/13_Validazione_Statistica.png")
571 plt.close()
572
573 # ============================================================
574 # 7. RIEPILOGO FINALE - VALUTAZIONE DEFINITIVA SU TEST SET
575 # ============================================================
576 print("\n[7/7] Riepilogo Risultati...")
577
578 print("\n    === CONFRONTO MODELLI (Test Set - Valutazione
    Finale) ===")
579 for name, model in trained_models.items():
580     y_p  = model.predict(X_test)
581     y_pr = model.predict_proba(X_test)[:, 1]
582     fpr, tpr, _ = roc_curve(y_test, y_pr)
583     print(f"    {name:25s} | "
584           f"Acc: {accuracy_score(y_test, y_p):.4f} | "
585           f"AUC: {auc(fpr, tpr):.4f} | "
586           f"Precision: {precision_score(y_test, y_p):.4f} | "
587           f"Recall: {recall_score(y_test, y_p):.4f}")
588
589 print(f"\n    === MITIGAZIONE BIAS (Test Set) ===")
590 print(f"    Gap Originale: {gap_bias:.4f} | Gap Mitigato: {
    gap_fair:.4f}")
591 print(f"    Riduzione: {((gap_bias - gap_fair) / gap_bias *
    100):.1f}%")
592
593 print(f"\n PIPELINE COMPLETATA. Grafici salvati in: '{
    OUTPUT_DIR}'")
```

Listing A.1: Pipeline completa di analisi — Barone.py

Bibliografia

- Angwin, J., Larson, J., Mattu, S., & Kirchner, L. (2016). Machine bias. *ProPublica*. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing

- Barocas, S., & Selbst, A. D. (2016). Big data's disparate impact. *California Law Review*, 104(3), 671–732. https://www.cs.yale.edu/homes/jf/BarocasSelbst.pdf

- Barocas, S., Hardt, M., & Narayanan, A. (2019). *Fairness and Machine Learning: Limitations and Opportunities*. https://fairmlbook.org

- Becker, B., & Kohavi, R. (1996). Adult. *UCI Machine Learning Repository*. https://doi.org/10.24432/C5XW20

- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. https://doi.org/10.1023/A:1010933404324

- Buolamwini, J., & Gebru, T. (2018). Gender shades: Intersectional accuracy disparities in commercial gender classification. *Proceedings of the 1st Conference on Fairness, Accountability and Transparency (FAT*)*, 77–91. https://proceedings.mlr.press/v81/buolamwini18a.html

- CBS News. (2015). Google apologizes for mis-tagging photos of African Americans as "gorillas". https://www.cbsnews.com/news/google-photos-labeled-pics-of-african-americans-as-gorillas/

- Chouldechova, A. (2017). Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. *Big Data*, 5(2), 153–163. https://arxiv.org/abs/1703.00056

- Cox, D. R. (1958). The regression analysis of binary sequences. *Journal of the Royal Statistical Society: Series B (Methodological)*, 20(2), 215–242. https://academic.oup.com/jrsssb/article/20/2/215/7027376

- Crenshaw, K. (1989). Demarginalizing the intersection of race and sex: A Black feminist critique of antidiscrimination doctrine, feminist theory and antiracist politics. *University of Chicago Legal Forum*, 1989(1), 139–167. https://chicagounbound.uchicago.edu/uclf/vol1989/iss1/8

- Dastin, J. (2018). Amazon scraps secret AI recruiting tool that showed bias against women. *Reuters*. https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G

- European Parliament. (2024). Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (AI Act). *Official Journal of the European Union*. https://eur-lex.europa.eu/eli/reg/2024/1689/oj

- Flores, A. W., Bechtel, K., & Lowenkamp, C. T. (2016). False positives, false negatives, and false analyses: A rejoinder to "Machine bias". *Federal Probation*, 80(2), 38–46. https://www.crj.org/publication/false-positives-false-negatives-false-analyses-rejoinder

- Friedman, B., & Nissenbaum, H. (1996a). Bias in computer systems. *ACM Transactions on Information Systems (TOIS)*, 14(3), 330–347. https://doi.org/10.1145/230538.230561

- Friedman, B., & Nissenbaum, H. (1996b). Accountability in a computerized society. *Science and Engineering Ethics*, 2(1), 25–42. https://doi.org/10.1007/BF02639315

- Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5), 1189–1232. https://doi.org/10.1214/aos/1013203451

- Geyik, S. C., Amber, S., & Kenthapadi, K. (2019). Fairness-aware ranking in search & recommendation systems with application at LinkedIn. *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2221–2231. https://doi.org/10.1145/3292500.3330691

- Glassdoor. (2015). Glassdoor's 2015 recruiting metrics revealed. https://www.glassdoor.com/employers/blog/glassdoors-2015-recruiting-metrics-revealed/

- Hardt, M., Price, E., & Srebro, N. (2016). Equality of opportunity in supervised learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 29. https://arxiv.org/abs/1610.02413

- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction* (2nd ed.). Springer. https://hastie.su.domains/ElemStatLearn/

- IMD Business School. (2018). Amazon's sexist hiring algorithm could still be better than a human. https://www.imd.org/research-knowledge/digital/articles/amazons-sexist-hiring-algorithm-could-still-be-better-than-a-human

- Kleinberg, J., Mullainathan, S., & Raghavan, M. (2016). Inherent trade-offs in the fair determination of risk scores. *arXiv:1609.05807*. https://arxiv.org/abs/1609.05807

- Lambrecht, A., & Tucker, C. (2019). Algorithmic bias? An empirical study of apparent gender-based discrimination in the display of STEM career ads. *Management Science*, 65(7), 2966–2981. https://doi.org/10.1287/mnsc.2018.3093

- LinkedIn Talent Solutions. (2019). *Gender Insights Report: How women and men find jobs differently*. Report ufficiale LinkedIn.

- McKinsey & Company. (2023). *Diversity matters even more: The case for holistic impact*. https://www.mckinsey.com/featured-insights/diversity-and-inclusion/diversity-matters-even-more-the-case-for-holistic-impact

- Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2021). A survey on bias and fairness in machine learning. *ACM Computing Surveys*, 54(6), 1–35. https://arxiv.org/abs/1908.09635

- Obermeyer, Z., Powers, B., Vogeli, C., & Mullainathan, S. (2019). Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366(6464), 447–453. https://www.science.org/doi/10.1126/science.aax2342

- OECD. (2024). Gender equality and work. https://www.oecd.org/gender/data/

- O'Neil, C. (2016). *Weapons of Math Destruction: How Big Data Increases Inequality and Threatens Democracy*. Crown.

- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12, 2825–2830. https://jmlr.org/papers/v12/pedregosa11a.html

- Rawls, J. (1971). *A Theory of Justice*. Harvard University Press.

- Select Software Reviews. (2025). 100+ recruitment statistics you need to know in 2025. https://www.selectsoftwarereviews.com/blog/applicant-tracking-system-statistics

- Simonite, T. (2018). When it comes to gorillas, Google Photos remains blind. *Wired*. https://www.wired.com/story/when-it-comes-to-gorillas-google-photos-remains-blind/

- *State v. Loomis*, 881 N.W.2d 749 (Wis. 2016). https://law.justia.com/cases/wisconsin/supreme-court/2016/2015ap000157-cr.html

- Verma, S., & Rubin, J. (2018). Fairness definitions explained. *Proceedings of the IEEE/ACM International Workshop on Software Fairness (FairWare'18)*. https://people.ece.ubc.ca/~mjulia/publications/Fairness_Definitions_Explained_2018.pdf

- World Economic Forum. (2023). *Global Gender Gap Report 2023*. https://www.weforum.org/publications/global-gender-gap-report-2023/