Alma Mater Studiorum · Università di Bologna

SCUOLA DI SCIENZE

Corso di Laurea Triennale in Informatica

Rilevamento della qualità del manto stradale: un approccio sperimentale tramite Machine Learning

Relatore:
Dott.
Federico Montori

Presentata da:
EMANUELE VISCONTI

Sessione IV
Anno Accademico 2020-21

Alla mia famiglia

Sommario

Il riconoscimento delle condizioni del manto stradale partendo esclusivamente dai dati raccolti dallo smartphone di un ciclista a bordo del suo mezzo è un ambito di ricerca finora poco esplorato. Per lo sviluppo di questa tesi è stata sviluppata un'apposita applicazione, che combinata a script Python permette di riconoscere differenti tipologie di asfalto. L'applicazione raccoglie i dati rilevati dai sensori di movimento integrati nello smartphone, che registra i movimenti mentre il ciclista è alla guida del suo mezzo. Lo smartphone è fissato in un apposito holder fissato sul manubrio della bicicletta e registra i dati provenienti da giroscopio, accelerometro e magnetometro. I dati sono memorizzati su file CSV, che sono elaborati fino ad ottenere un unico DataSet contenente tutti i dati raccolti con le features estratte mediante appositi script Python. A ogni record sarà assegnato un cluster deciso in base ai risultati prodotti da K-means, risultati utilizzati in seguito per allenare algoritmi Supervised. Lo scopo degli algoritmi è riconoscere la tipologia di manto stradale partendo da questi dati. Per l'allenamento, il DataSet è stato diviso in due parti: il training set dal quale gli algoritmi imparano a classificare i dati e il test set sul quale gli algoritmi applicano ciò che hanno imparato per dare in output la classificazione che ritengono idonea. Confrontando le previsioni degli algoritmi con quello che i dati effettivamente rappresentano si ottiene la misura dell'accuratezza dell'algoritmo.

Introduzione

In questa tesi si è voluto studiare come l'applicazione di opportuni algoritmi di machine learning possano fornire indicazioni rilevanti sulla qualità della superficie stradale e individuare talune criticità al fine di migliorare l'esperienza d'uso delle strade stesse. Sono stati dapprima definiti gli strumenti e la metodologia di lavoro ritenuti idonei a questa ricerca, successivamente è stata implementata l'applicazione per la raccolta dei dati. I dati a disposizione sono stati elaborati tramite script fino ad ottenere dei modelli di Machine Learning in grado di riconoscere differenti conformazioni di asfalto. Tutti i passaggi seguiti saranno spiegati nella presente tesi. Nel primo capitolo si tratterà lo stato dell'arte, le implicazioni di questo campo di ricerca e si faranno dei cenni a lavori finora svolti in questo ambito. Nel secondo capitolo verrà illustrata l'architettura di questo lavoro di ricerca e la procedura seguita per la raccolta dei dati necessari a raggiungere l'obiettivo. Nel terzo capitolo saranno esposti il processo di estrazione ed elaborazione dei dati, i risultati preliminari ottenuti e l'analisi effettuata per la selezione delle features. Nel quarto capitolo verrà trattato l'approccio finale utilizzato per la classificazione dei tratti stradali con i relativi risultati. Nell'ultimo capitolo saranno analizzati i risultati ottenuti.

Indice

Elenco delle figure

Capitolo 1

Stato dell'arte

1.1 Introduzione al Machine Learning

Con il termine Machine Learning (apprendimento automatico) si intende il sottoinsieme di attività svolte da una AI (Intelligenza artificiale) che fornisce alle macchine la possibilità di migliorare e apprendere dai propri errori, così da accumulare esperienza, senza la necessità di essere riprogrammate. Questi algoritmi, capaci di apprendere ed accumulare esperienza, hanno rivoluzionato il compito del programmatore. Essi per funzionare hanno bisogno di un approccio strutturato che consenta l'apprendimento automatico.

Metodi di Machine Learning

In base alle modalità con le quali avviene il processo di apprendimento di un modello di Machine Learning, è possibile distinguere due metodologie principali: Supervised Learning e Unsupervised Learning. Nel Supervised Learning l'algoritmo necessita di una classificazione dei dati forniti in input per poter in seguito classificare dei nuovi dati non etichettati. Un uso comune del Supervised Learning è il caso della classificazione delle immagini, dove l'algoritmo viene alimentato da dati con immagini etichettate col nome dell'oggetto rappresentato e sarà successivamente in grado di riconoscere altre immagini dello stesso oggetto. Nel caso di Unsupervised Learning l'algoritmo

lavora con dati non etichettati, cercando in autonomia di riconoscere degli schemi all'interno del set di dati, senza ricevere feedback di correzione sulle scelte effettuate; un contesto in cui gli algoritmi di Unsupervised Learning trovano utilizzo sono i sistemi di raccomandazione nell'e-commerce, dove ad un utente vengono suggeriti degli articoli acquistati da altri utenti aventi gusti simili.

Modelli di Machine Learning

La scelta del modello da utilizzare è fortemente legata al problema da risolvere. Esistono diversi modelli, i più diffusi sono:

- Clustering: mediante l'analisi dei dati raccolti permette la suddivisione in sottoinsiemi, detti cluster. Questi cluster sono accomunati da criteri di similitudine, che possono essere singoli o molteplici.
- Alberi di decisione: lo scopo dell'albero di decisione è un modello di predizione, che consente di mappare i dati riguardanti un'attività e determinare delle conclusioni riguardanti la stessa.
- ILP (Programmazione logica induttiva): è un modello di programmazione per esempi. Partendo da una conoscenza di base e dal set di esempi logici, permette di produrre una programmazione logica capace di selezionare gli esempi positivi da quelli negativi.
- Deep learning: esso si basa sull'utilizzo di reti neurali composte su più livelli. È un approccio moderno che trae ispirazione dal funzionamento della rete neurale organica, per questo motivo ha trovato utile applicazione per lo sviluppo di protesi artificiali avanzate.

Lo studio e l'implementazione di questi algoritmi, che permettono l'apprendimento di informazioni a partire da dati disponibili, è fortemente legato all'impiego in quei sistemi con un problema di computazione che non consente una programmazione classica rendendola impraticabile o poco conveniente. Questo significa accantonare la classica programmazione esplicita, sistema in

cui la componente umana programma un modello in base a comandi del tipo "if-then", a favore di un metodo in cui la macchina è in grado di stabilire da sola gli schemi da seguire per ottenere il risultato desiderato. Pertanto, il vero fattore che distingue l'intelligenza artificiale è l'autonomia. Volendo generalizzare, è possibile scindere un processo qualunque di Machine Learning in tre fasi principali:

- l'apprendimento dei dati, nelle diverse forme in cui si possono manifestare;
- la valutazione dei dati stessi, nella quale il sistema informatico ipotizza dei modelli statistici che descrivano la realtà osservata;
- l'ottimizzazione dei modelli stimati e la conseguente formulazione di una strategia di risposta/azione in base ai feedback raccolti con l'esperienza.

1.2 Human Activity Recognition con smartphone

Lo scopo della Human Activity Recognition(HAR) è l'analisi e la comprensione delle attività umane partendo da segnali acquisiti da dispositivi indossabili ed ambientali, quali possono essere videocamere, accelerometri e giroscopi[9]. Quello della HAR è un ramo di ricerca che ha visto negli ultimi anni un incremento di popolarità. Tale successo è riconducibile al vasto impiego che questa tecnica può avere, spaziando dalla realizzazione di sistemi ludico-ricreativi quali videogiochi fino al suo utilizzo in ambiti sociologici più importanti quali possono essere il crowd control, l'healthcare support (come ad esempio un sistema di fall detection per anziani volto ad avvisarne i familiari), il lifestyle monitoring e l'interazione-uomo macchina. In letteratura è possibile riconoscere due approcci predominanti alla Human Activity Recognition: il primo viene detto Sensor-based e si basa principalmente sul-

l'impiego di sensori indossabili da un individuo al fine di registrarne i movimenti; il secondo è di tipo Vision-based, e prevede l'utilizzo di videocamere per estrapolare informazioni sull'azione in corso.

A prescindere dall'approccio scelto, gli step generalmente previsti per un processo di Human Activity Recognition fanno riferimento ad un protocollo denominato Activity Recognition Process (ARP). ARP consiste nell'esecuzione di 5 step: data acquisition, preprocessing, data segmentation, feature extraction e classification. Lo step di data acquisition si occupa del rilevamento dei dati registrati dai sensori in uso. La fase di preprocessing consiste nella rimozione del rumore dei segnali rilevati dai sensori e dall'eliminazione di valori o componenti non ritenuti idonei all'elaborazione Nello step di data segmentation i dati vengono suddivisi in blocchi, altrimenti chiamati windows; la prassi comunemente più utilizzata nell'Activity Recognition prevede la divisione dei dati in blocchi di lunghezza fissa(sliding window)[1]. Con lo step di features extraction si identifica un processo matematico attraverso il quale partendo dall'insieme di dati iniziale, si ottiene un nuovo insieme di features attraverso una funzione di mapping.

Ad esempio, con $x = \{x_1, x_2, ..., x_n\} \in R_n$ rappresentante un insieme di dati, una feature f_i è data da $f_i = g_i \{x_1, x_2, ..., x_n\}$ con i = 1,..., n, dove $g_i : R_n \to R$ è una funzione di mapping. L'ultimo step prevede l'allenamento di algoritmi di Machine Learning, attraverso i quali si otterranno dei modelli che verranno successivamente testati per appurarne l'accuratezza in fase di riconoscimento e classificazione delle attività.

In ambito Sensor-based i sensori maggiormente adoperati nella fase di acquisizione dati sono accelerometro e giroscopio, ed entrambi fanno ormai parte dell'hardware standard previsto nella fabbricazione degli smartphone prodotti al giorno d'oggi. La loro larga diffusione e il costante utilizzo, trasversali alle varie fasce d'età, rende quindi gli smartphone particolarmente interessanti ed utili dal punto di vista della ricerca nel campo della HAR, in quanto questi ultimi non vengono percepiti dagli individui che li utilizzano come ulteriori elementi di ingombro poiché strettamente integrati nella rou-

tine quotidiana[4]. I più recenti studi per la Human Activity Recognition sono svolti adoperando smartphone Android[7]. Le API messe a disposizione da Android forniscono letture da sensori sia di tipo fisico che di tipo virtuale; nel caso dei sensori virtuali, i segnali sono calcolati dal sistema processando ed occasionalmente fondendo segnali provenienti da sensori fisici, come avviene con la rimozione della componente gravitazionale nelle letture dall'acceleratore lineare.

1.3 Road quality

Un altro campo in cui il Machine Learning sta trovando largo impiego e sviluppo è la gestione della "Road quality". Il monitoraggio della strada e delle sue condizioni, come ad esempio, stato del manto stradale, informazioni meteo e congestione del traffico, sta diventando ogni giorno più importante. Data le varie tipologie di strade, è necessaria una tecnologia che sia capace di analizzare e identificare le anomalie tramite un'analisi efficiente dei dati registrati con l'ausilio di appositi sensori che monitorino. Le diverse tipologie di veicoli e di condizioni stradali che possono presentarsi, rendono necessaria l'implementazione di tecnologie che riescano a individuare eventuali anomalie e che siano capaci di elaborare i dati, così da strutturare una risposta attiva.

I principali obiettivi della "Road quality" sono quindi quelli di migliorare l'esperienza di guida e ridurre gli incidenti che possono causare danni ai veicoli oppure il ferimento o nei casi più gravi la morte delle persone. Una costante attività di monitoraggio permette quindi di avvisare le autorità competenti, nel caso vengano rilevate delle criticità, e consentire delle azioni di correzione.

I sistemi tradizionali di monitoraggio non consentono di verificare la qualità dell'intero percorso stradale ma soltanto di alcune sue parti, o con frequenza di aggiornamento non ottimale. Inoltre si deve tener conto del fatto che spesso le rilevazioni vengono effettuate manualmente e questo presenta degli aspetti negativi:

- Alti costi di manodopera

- Difficoltà nell'effettuare un'attività preventiva, piuttosto la criticità viene rilevata solo nel momento in cui si verifica
- Scarsa oggettività nelle rilevazioni

Sono stati proposti altri metodi di monitoraggio che però comportano elevati costi di installazione della strumentazione idonea nonché regolare manutenzione della stessa che fa lievitare ulteriormente i costi. In questo contesto un nuovo tipo di approccio basato sull'impiego di dispositivi mobile combinato all'uso di sensori quali accelerometro, giroscopio, e GPS ha fornito risultati promettenti ed il lavoro di tesi è stato incentrato su quest'ultimo approccio.

1.4 Motivazioni

In letteratura esistono già degli studi sul grado di comfort di un ciclista su strada, alcuni di questi utilizzano come metro di misura alcuni fattori ambientali o prettamente urbani come volume e velocità del traffico, ampiezza del percorso ciclabile e attività commerciali a bordo strada[10], spesso utilizzando nell'allenamento dei modelli di Machine Learning i dati raccolti tramite videocamera e i dati raccolti da sensori di movimento montati sulla bicicletta. Altri studi hanno dimostrato la possibilità di riuscire a distinguere un terreno asfaltato da uno non asfaltato utilizzando il GPS di uno smartphone[8, 3]. A mia conoscenza, esiste solo uno studio svolto con l'obiettivo di riuscire a classificare diverse tipologie di manto stradale ricorrendo unicamente ai dati rilevabili per mezzo di uno smartphone; la ricerca condotta da Hoffman,Mock e May riporta la possibilità di riconoscere tre tipologie di terreno, nel loro studio denominate smooth, bumby e rough[6]. Tuttavia, il loro lavoro si basa su un approccio totalmente Supervised con labeling manuale. Il lavoro di questa tesi propone un percorso alternativo a quello da loro intrapreso, cercando un approccio misto Supervised-Unsupervised per dare una risposta al problema della classificazione delle condizioni del manto stradale partendo esclusivamente dai dati acquisiti da smartphone

Capitolo 2

Implementazione App e raccolta dati

Per reperire i dati necessari all'allenamento dei modelli di Machine Learning è stata sviluppata un'apposita applicazione per dispositivi Android, denominata RidingStyle. I dati raccolti consistono nei valori acquisiti dai sensori di movimento di uno smartphone mentre l'utente è alla guida di una bicicletta, con lo smartphone fissato in un apposito holder sul manubrio. In particolare, si è scelto di rilevare i dati provenienti da accelerometro, giroscopio, magnetometro e accelerometro lineare, insieme alla posizione GPS in cui i dati sono stati rilevati.

2.1 App Overview

L'applicazione prevede l'inserimento di uno o più utenti a cui i dati raccolti faranno riferimento. Selezionando l'utente in carico della raccolta dati, verrà presentata la possibilità di decidere il tipo di vista da utilizzare durante il tragitto; sarà possibile scegliere tra l'utilizzo di una mappa o una schermata per l'acquisizione video.

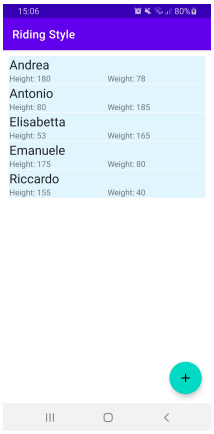

Figura 2.1: Elenco utenti

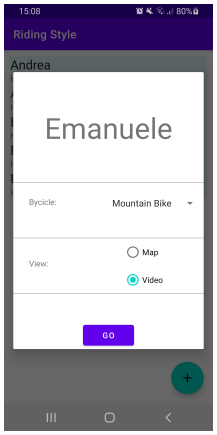

Figura 2.2: Selezione schermata

I dati provenienti dai sensori di movimento e dal GPS inizieranno ad essere rilevati nel momento in cui l'utente premerà il bottone per avviare la registrazione del video e si interromperà quando l'utente premerà nuovamente il bottone per interromperla; lo stesso principio di funzionamento è implementato per la schermata contenente la mappa, dove l'utente avvierà ed interromperà l'acquisizione dei dati mediante l'apposito bottone di Start/Stop.

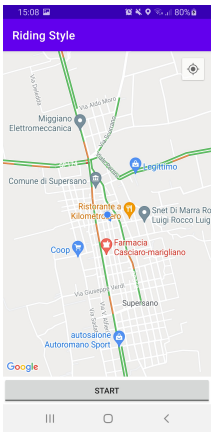

Figura 2.3: Map view

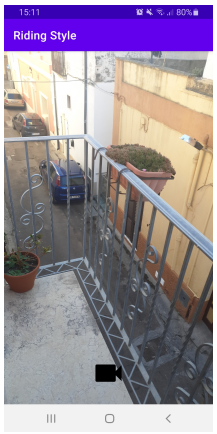

Figura 2.4: Video view

2.2 Scelte progettuali

Si è scelto di prevedere l'utilizzo di più utenti sullo stesso smartphone per avere la possibilità di raccogliere dati da ciclisti fisicamente differenti, mantenendo la sensibilità dello stesso dispositivo. La mappa, sebbene non di fondamentale importanza ai fini della raccolta dati, è stata implementata per personale utilità in caso di raccolta dati in aree non familiari. L'acquisizione dei video relativi al tragitto effettuato ha lo scopo di fornire un successivo riscontro visuale sui dati raccolti; tale riscontro è volto all'etichettatura dei dati che saranno utilizzati per allenare gli algoritmi di Supervised Learning. Per ogni sensore di movimento sono stati acquisiti i valori relativi alle assi x,y e z. La ragione di questa scelta è da ricercarsi nelle migliori performance offerte dai modelli di Machine Learning allenati con features provenienti da tutte e tre le assi dei sensori[2].

2.3 Implementation details

1. Database

   Il database è implementato mediante l'utilizzo di RoomDatabase, le Entità previste sono:

   `Rider (id, name, weight, height)`: si riferisce agli utenti che utilizzeranno l'applicazione.

   `Route (id, rider_name, bike_type, date)`: si riferisce ai percorsi effettuati da un particolare utente in una certa data.

   Ogni entità ha il corrispettivo ViewModel, Repository ed interfaccia Dao usati per l'esecuzione delle operazioni su database da parte dei Fragment che operano con le entità Rider e/o Route.

2. Lettura dati dai sensori

   I sensori scelti per effettuare la raccolta dati sono accelerometro, giroscopio, accelerometro lineare e magnetometro, unitamente ai dati sulla

posizione acquisiti tramite GPS mediante FusedLocationProviderClient con accuratezza settata a `LocationRequest.PRIORITY_HIGH_ACCURACY`.

In fase di registrazione del listener tramite SensorManager, la velocità di campionamento è settata a SensorManager.`SENSOR_DELAY_FASTEST`, questo per ottenere la maggior quantità di dati possibile, inoltre viene fornito il riferimento ad un HandlerThread in cui il metodo onSensorChanged(SensorEvent sensorEvent) verrà chiamato e da qui verrà effettuata la scrittura su file CSV ad ogni aggiornamento ricevuto; ogni scrittura sul file CSV trascriverà sia i valori del sensore appena aggiornato sia i valori dei sensori non aggiornati memorizzati in precedenza (es. se sono stati rilevati aggiornamenti da parte dell'accelerometro, verranno trascritti i nuovi valori di accelerometro insieme agli ultimi valori di giroscopio, magnetometro e accelerometro lineare rilevati in precedenza). I valori dei sensori campionati sono inseriti insieme al relativo timestamp;

Il nome del file CSV avrà forma:

```
RouteID<routeID>__acc_lacc_gyr_mag_gps_timestamp_
<user>_<bike_type>_.csv
```

e ogni riga avrà forma:

```
<acc_x>,<acc_y>,<acc_z>,<lacc_x>,<lacc_y>,<lacc_z>,<gyr_x>,
<gyr_y>,<gyr_z>, <mag_x>,<mag_y>,<mag_z>,<latitude>,
<longitude>,<timestamp>
```

3. Mappa

   La mappa è implementata tramite:

```
<fragment
xmlns:android="http://schemas.android.com/apk/res/android"
android:id="@+id/map"
android:name="com.google.android.gms.maps.SupportMapFragment"/>
```

ed inizializzata nel relativo Fragment tramite GoogleMap. Per l'acquisizione della posizione si è utilizzato FusedLocationProviderClient, e l'aggiornamento su mappa della posizione avviene in tempo reale ad ogni chiamata della callback LocationCallback().

4. Video

   Per la registrazione video, è stata utilizzata l'API camera2 messa a disposizione da Android in congiunzione a MediaRecorder; le sessioni di cattura video avvengono in un thread separato. Nella nomenclatura del file è incluso l'ID del tragitto a cui è associato il video, di modo che in fase di labeling dei dati si possa sapere quali dati raccolti corrispondano a quale condizione del manto stradale; il nome del file avrà quindi forma:

   `ROUTEID_<routeID>___<yyyyMMdd__HHmmSS>.mp4`

2.4 Testing devices

L'applicativo è stato testato su Samsung S9 con Android 10. La bicicletta utilizzata è stata una mountain-bike Atala con ammortizzatori a molla su forcelle e telaio. Al fine di raccogliere i dati nella maniera più omogenea possibile, la posizione dell'holder per smartphone è stata la medesima per tutte le sessioni di raccolta dati.

2.5 Plot dei DataPoints

L'ammontare dei dati raccolti costituisce un DataSet di 6500 record, dove ogni record contiene features calcolate su una finestra di tempo di 3 secondi. I punti in blu sulla seguente mappa rappresentano tutti i data points raccolti finora e contenuti nel DataSet. É stato mappato l'intero territorio di Supersano(LE).

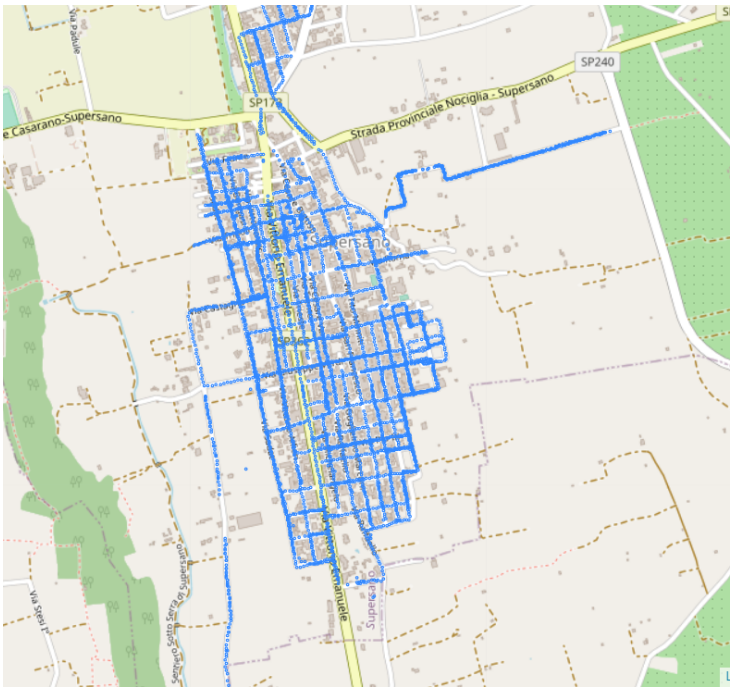

Figura 2.5: Mappa di Supersano

Capitolo 3

Estrazione dati ed elaborazione

I processi di estrazione ed elaborazione dei dati, allenamento e testing dei modelli sono stati eseguiti off-line su di un normale PC.

Linguaggio utilizzato: Python

Librerie: sklearn,numpy,geopy,pandas,seaborn

3.1 Estrazione dati

Step1: Data Cleaning

Al termine di ogni sessione di raccolta dati, i CSV contenenti i dati raccolti dai sensori vengono caricati sotto forma di DataFrame (Pandas), e subiscono un pre-processo di Data Cleaning al fine di rimuovere valori nulli e/o non significativi (es. NaN, timestamp nullo).

Step2: Calcolo Magnitude

Per ogni riga dei DataFrame viene calcolata la magnitude relativa ad ogni sensore sulla base dei suoi valori sulle tre assi x,y,z:

$$m = \sqrt{x^2 + y^2 + z^2}$$

Si otterranno quindi dei nuovi DataFrame dove ogni riga conterrà i valori di magnitude dei 4 sensori scelti, mantenendo i dati sulla posizione GPS e il timestamp.

Step3: Features Extraction

Ogni DataFrame così ottenuto verrà ulteriormente elaborato al fine di estrarre delle features sulla base di una prestabilita finestra di tempo, determinata a partire dai timestamp contenuti nelle righe del DataFrame. L'ampiezza scelta per finestra di tempo è di 3 secondi, valore che in letteratura risulta essere tra i più utilizzati[5] e ritenuto adatto a raccogliere le informazioni rilevabili nell'arco di un'intera pedalata. Le features estratte per ogni sensore sono: varianza, media, deviazione standard, minimo, massimo, range, skewness e kurtosis. Le rispettive abbreviazioni riportate successivamente in questa tesi sono: VAR,AVG,STD,MIN,MAX,RNG,SKW,KUR. Di conseguenza, per indicare ad esempio la varianza relativa all'accelerometro lineare si userà la dicitura LACC_VAR

I DataFrame prodotti da questa elaborazione saranno quindi composti da 34 colonne: 2 colonne per i valori di latitudine e longitudine, 8 colonne per le features del giroscopio, 8 colonne per le features del magnetometro, 8 colonne per le features dell'accelerometro e 8 colonne per le features dell'accelerometro lineare. Tutti i DataFrame contenenti le features finora prodotti, verranno concatenati per formare infine un unico DataFrame relativo alla sessione di raccolta dati appena conclusa.

Step4: Features Scaling

In fase di allenamento di un modello di Machine Learning, il DataSet fornito in input consisterà nella concatenazione dei DataFrame di ogni sessione, con tutte le features opportunamente scalate mediante StandardScaler.

Il processo finora descritto fa riferimento alle sessioni di raccolta dati in cui non è previsto un labeling dei dati, in quanto sessioni volte a fornire un insieme di allenamento ad un algoritmo di apprendimento non supervisionato (K-means) Nelle sessioni di raccolta volte ad allenare algoritmi di Supervised Learning, ai dati contenuti nei CSV sono preventivamente applicate delle label per assegnare una classe di appartenenza ai dati in questione, dove tale label viene scelta previa visione del video registrato e associato al percorso i cui dati nei CSV fanno riferimento. Le label assegnate ai dati saranno

mantenute nei vari passaggi descritti in precedenza, dal Data Cleaning fino al Feature Scaling. Almeno in fase iniziale, la scelta delle label previste per le condizioni dell'asfalto è ricaduta su "BuoneCondizioni" e "CattiveCondizioni, ovvero le due tipologie di manto stradale distinguibili con maggior chiarezza da un osservatore umano.

3.2 Risultai preliminari

3.2.1 SUPERVISED RESULTS

I primi test condotti hanno avuto come oggetto un DataSet con dati classificati manualmente, di modo che potesse essere utilizzato per allenare algoritmi di Supervised Learning. Le label sono scelte visionando i video associati ai percorsi cui i dati fanno riferimento. In questa fase la classificazione dei dati prevede la suddivisione degli stessi in due categorie identificate dalle seguenti due label:

1. "CattiveCondizioni" - il manto stradale presenta forti irregolarità quali buche o crepe

2. "BuoneCondizioni" - il manto stradale risulta omogeneo e regolare

Gli algoritmi testati sono: SupportVectorMachine(SVM), StochasticGradient-Descent(SGD) e RandomForest(RF). In questa fase sono state utilizzate tutte le features inizialmente previste ed estratte.

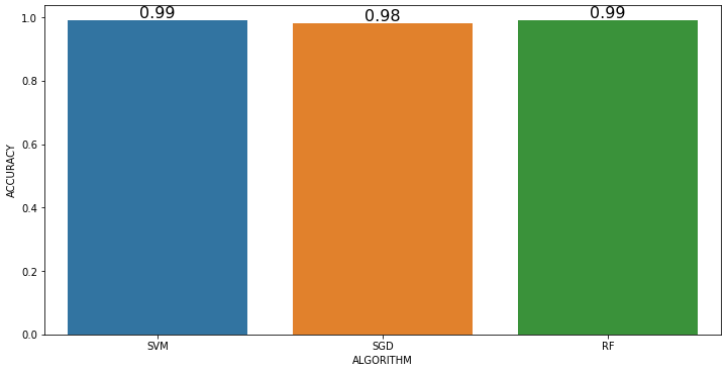

Figura 3.1: Accuracy con 2 classi

Come si evince dal grafico, l'accuratezza risulta elevata. Tale precisione è in linea con i risultati di prove empiriche effettuate tramite visionando i video associati a DataSet di prova (non contenuti nei training-set e test-set precedentemente adoperati) sui quali sono stati testati i classificatori ottenuti.

3.2.2 UNSUPERVISED RESULTS

Dopo aver appurato che mediante Supervised Learning sia effettivamente possibile classificare con elevata accuratezza le due tipologie di manto stradale distinguibili da un osservatore umano, si è deciso di indagare sulla possibilità di poter identificare un maggior numero di tipologie di superfici stradali adottando un approccio Unsupervised, in particolare ricorrendo al clustering tramite K-means.

Per verificare la correttezza di tale approccio, è stato prima controllato che un clustering a 2 cluster fornisse dei risultati comparabili a quelli ottenuti con il labeling manuale. Per far ciò, si è lanciato il K-means sul DataSet fornito in input agli algoritmi Supervised ovviamente rimuovendo preventivamente le label, e impostando a 2 il numero dei cluster voluti. In seguito è stata controllata per ognuno dei 2 cluster la percentuale dei dati precedentemente associati alle label 'BuoneCondizioni' e 'CattiveCondizioni'. I risultati sono riportati di seguito.

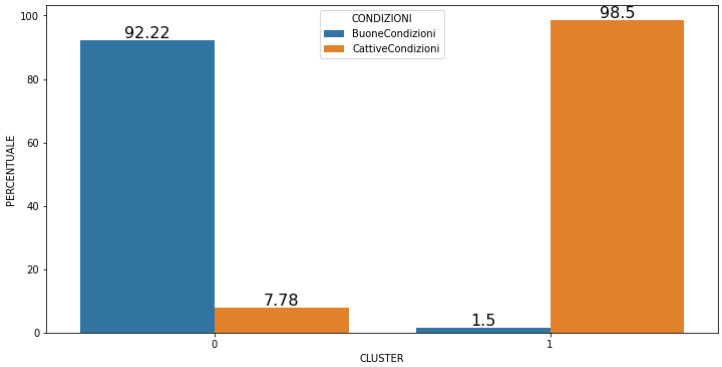

Figura 3.2: Percentuali di label per cluster

Come visibile nel grafico, i cluster assegnati da K-means sono sostanzialmente comparabili con la classificazione effettuata manualmente. Il clustering ottenuto da K-means può quindi essere considerato significativo.

3.3 N-CLUSTERING E ANOVA F-TEST

Per indagare su quante tipologie di manto stradale possano essere effettivamente distinte tramite clustering, si è dapprima lanciato il K-means per identificare nello stesso DataSet un numero di cluster maggiore rispetto ai 2 cluster di partenza; le prove di clustering sono state eseguite per un numero di cluster che va da 2 a 10.

Successivamente, è stato condotto uno studio mediante Anova F-Test allo scopo di identificare le migliori features tra quelle finora utilizzate nei diversi clustering.

3.3.1 ANOVA F-TEST

L'ANOVA (Analysis of Variance) è un test statistico che mira a riconoscere la presenza di differenze tra gruppi all'interno di un insieme di dati; il risultato del test è rappresentato da un valore numerico f-value (in letteratura noto anche come F-statistic) che fornisce un'indicazione su quanto determinati gruppi possano considerarsi separati.

L'ANOVA F-Test condotto ha quindi come obiettivo l'analisi dei dati contenuti nei DataSet suddivisi in cluster al fine di estrapolare per ogni feature un valore numerico che indichi quanto questa'ultima funga da discriminante tra i cluster presi in esame.

Analiticamente, la formula matematica per ottenere il valore f-value è:

$$F = \frac{MSB}{MSW} = \frac{SSb/(k-1)}{SSw/(N-k)}$$

dove

- $SSb = \sum_i n_i(\bar{y_i} - \bar{y})^2$: (sum of squares between groups): questo termine fornisce una misura della distanza tra i cluster, ovvero la distanza tra le medie nelle distribuzioni dei gruppi

- $SSw = \sum_{ij}(\bar{y_{ij}} - \bar{y_i})^2$: (sum of squares within groups): questo termine fornisce una misura sulla "compattezza" dei cluster in termini di varianza

Il valore di F tenderà quindi a salire in presenza di gruppi tra loro distanti e con bassa varianza interna, indicando come giusta l'ipotesi di considerare la feature in esame come una buona discriminante tra i gruppi. Se invece nessuna differenza reale è presente tra i cluster testati, ipotesi che in letterature viene chiamata Null Hypothesis, il valore di F ottenuto tenderà ad assumere valori vicini ad 1.

I seguenti grafici rappresentano gli f-value di ogni feature in relazione ai clustering ottenuti in precedenza.

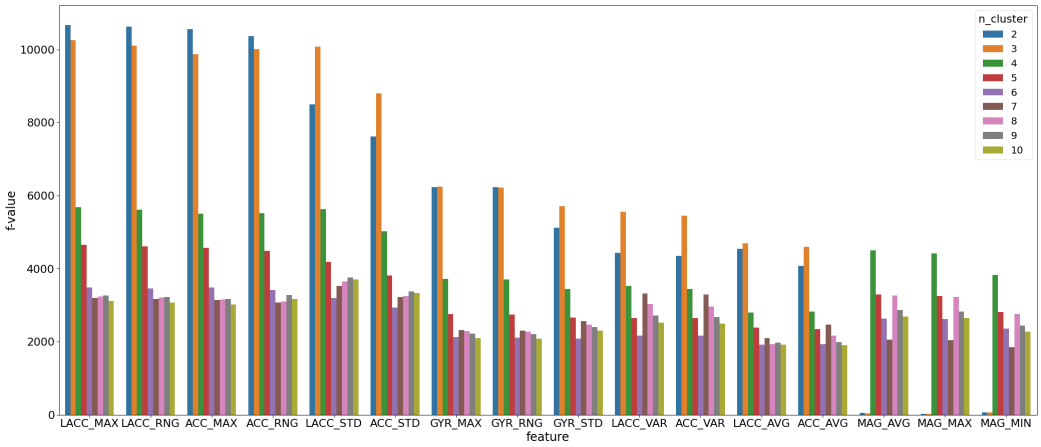

Figura 3.3: F-value delle migliori features

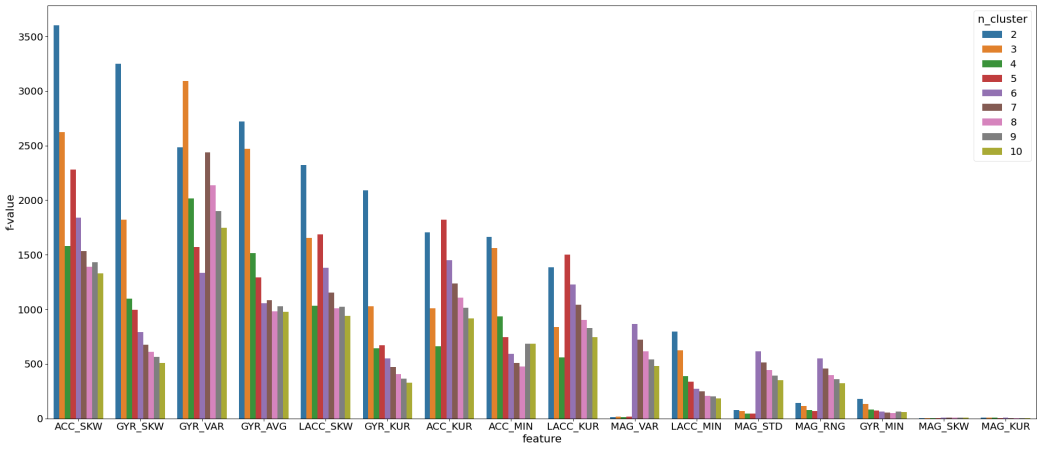

Figura 3.4: F-value delle peggiori features

Come si evince dai grafici, alcune feature presentano dei valori di f-value particolarmente elevati rispetto ad altre; inoltre è possibile notare come tale valore risulti migliore in presenza dei clustering a 2, 3 e 4 cluster. Conseguentemente ai risultati così ottenuti, i successivi test svolti hanno come oggetto i Dataset con raggruppamenti a 2, 3 e 4 cluster e prevedono una suddivisione delle features in due gruppi: il primo raggruppa le features considerate migliori e rappresentate nella metà sinistra del grafico, il secondo gruppo contiene le restanti features.

Il contenuto dei due gruppi sarà quindi:

- GoodFeatures: `LACC_MAX`, `LACC_RNG`, `ACC_MAX`, `ACC_RNG`, `LACC_STD`, `ACC_STD`, `GYR_MAX`, `GYR_RNG`, `GYR_STD`, `LACC_VAR`, `ACC_VAR`, `LACC_AVG`, `ACC_AVG`, `MAG_AVG`, `MAG_MAX`, `MAG_MIN`

- BadFeatures: `ACC_SKW`,`GYR_SKW`,`GYR_VAR`,`GYR_AVG`,`LACC_SKW`,`GYR_KUR`, `ACC_KUR`,`ACC_MIN`,`LACC_KUR`,`MAG_VAR`,`LACC_MIN`,`MAG_STD`, `MAG_RNG`, `GYR_MIN`,`MAG_SKW`,`MAG_KUR`

3.3.2 Accuracy test

Per validare ulteriormente i risultati ottenuti, è stato condotto un test con algoritmi Supervised fornendo come label i cluster ottenuti dal K-means e controllando il variare dell'accuracy in base all'impiego di tutte le features, poi di quelle buone ed infine quelle ritenute peggiori. I risultati sono riportati di seguito.

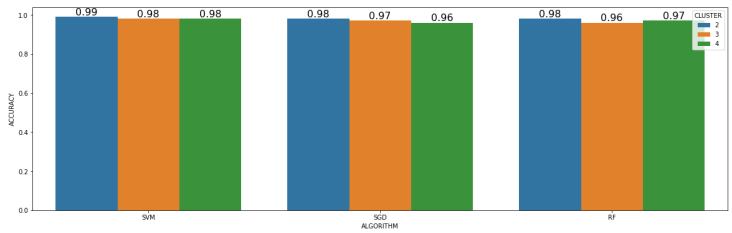

Figura 3.5: Accuracy con tutte le features

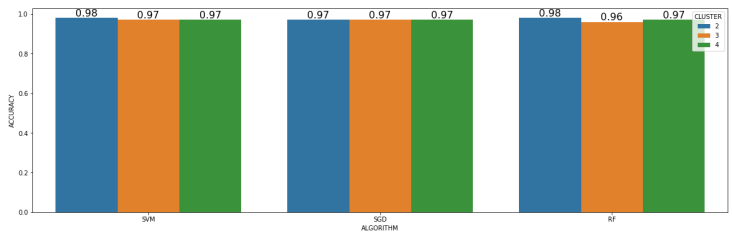

Figura 3.6: Accuracy con le migliori features

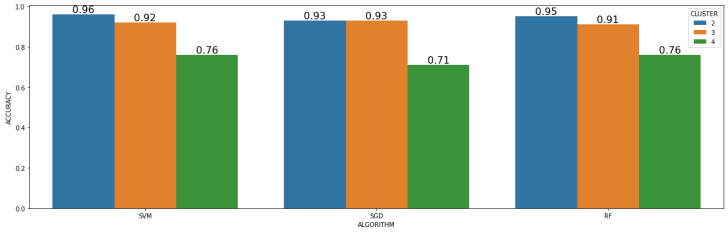

Figura 3.7: Accuracy con le peggiori features

Risultati

Si può vedere come i risultati ottenuti adoperando tutte le features e solo quelle migliori siano in linea di massima equivalenti; le accuracy inerenti all'utilizzo delle features ritenute peggiori, sebbene non evidenzino un drastico calo nel test con 2 cluster, mostrano una perdita di accuratezza vicina al 5% nel test a 3 cluster e vicina al 20% nel test a 4 cluster.

Considerazioni

Generalmente in un ANOVA test, si tende a rigettare la Null Hypothesis se il valore f-value ottenuto risulta minore ad un F-critical value determinato dai gradi di libertà e un valore di soglia alpha, dove alpha indica il livello di significatività del test e specifica la percentuale di rischio accettato nel rigettare la Null Hypothesis quando è invece vera. Nei risultati ottenuti gli unici f-value effettivamente minori a F-critical value erano inerenti alle feature di skewness e kurtosis del magnetometro. Questo dice che anche le features ritenute peggiori influiscono in qualche misura nella distinzione tra cluster, quindi la decisione sulle features ritenute migliori è stata presa principalmente basandosi sui valori di f-value più elevati.

Capitolo 4

Clustering

Per avere un riscontro visuale su come il territorio interessato dalla raccolta dati sia stato classificato, i data points sono stati disegnati su mappa variando il colore in base al cluster di appartenenza. Sono di seguito riportate 3 mappe, di cui la prima rappresenta i data points del DataSet con raggruppamento a 2 cluster, la seconda a 3 cluster e l'ultima con raggruppamento a 4 cluster.

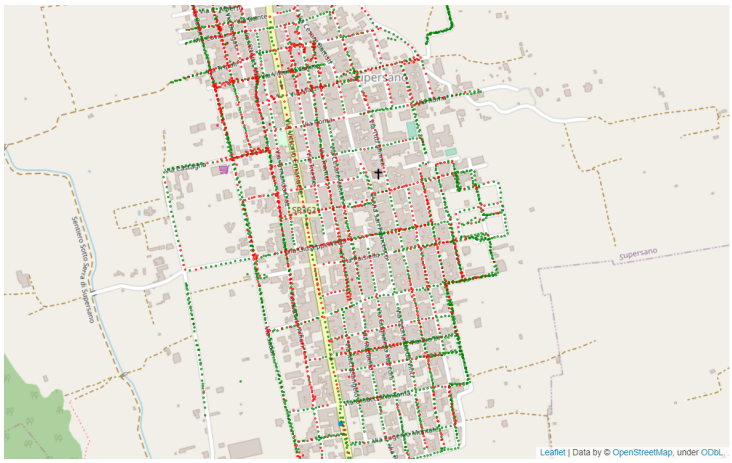

Figura 4.1: Raggruppamento a 2 cluster

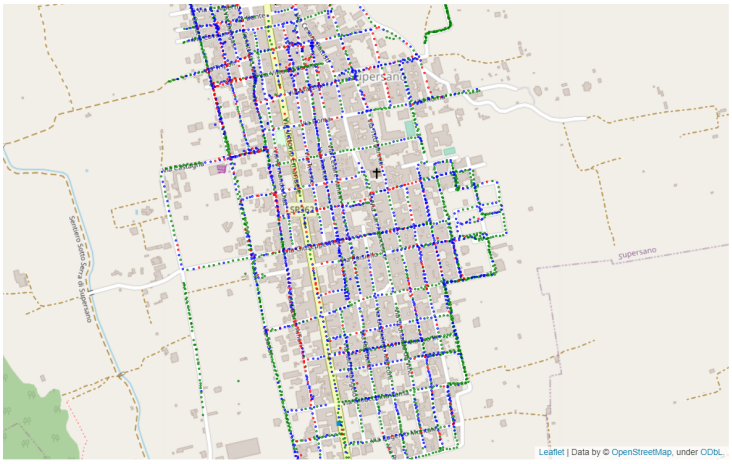

Figura 4.2: Raggruppamento a 3 cluster

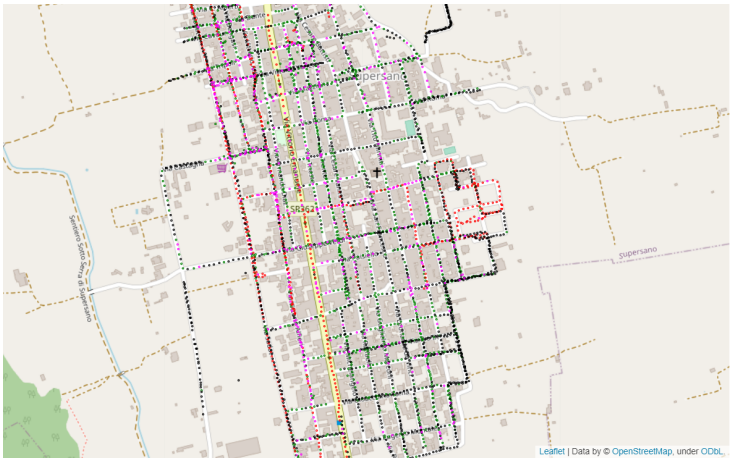

Figura 4.3: Raggruppamento a 4 cluster

Come appare evidente soprattutto nelle figure 4.3 e 4.4, è possibile distinguere su tratti di strada anche brevi la presenza di data points riferiti a cluster differenti; le ragioni sono da imputare a due fattori non mutualmente esclusivi:

- Conformazione del particolare tratto di strada: alcuni segmenti di strada possono avere delle condizioni dell'asfalto differenti tra le due corsie e/o in punti adiacenti; tali discrepanze possono essere state rilevate in

momenti diversi della raccolta dati, ad esempio ripassando sullo stesso tratto di strada ma percorrendolo in senso inverso.

- Imprecisione della classificazione: data la natura statistica del processo di assegnazione di una label da parte di un classificatore, è possibile che un data point possa essere stato classificato erroneamente.

Poiché in ultima istanza questo lavoro di tesi mira a fornire uno strumento volto a classificare le condizioni del manto stradale su di un tratto di strada, la presenza di data points dalla classificazione mista su di uno stesso segmento stradale risulta essere un elemento di confusione ai fini della classificazione del segmento stesso. La soluzione pensata per rispondere a questo problema è la seguente: suddividere una strada in tratti da 200m, ed associare a tutti i data points associati al quel tratto di strada il cluster che tra di essi risulta predominante. Le modifiche apportante in questo senso ai DataSet sono visibili sulle nuove mappe risultanti.

Figura 4.4: Data points di 2 cluster differenti

Figura 4.5: Data points con cluster predominante

4.0.1 Accuracy test con nuovi cluster

L'ultimo step è stato quello di testare l'accuratezza dei classificatori nel loro operare con i DataSet modificati secondo il criterio descritto in precedenza, vale a dire la loro di capacità di classificare i data points di un tratto di strada mantenendo un'accuratezza accettabile. Anche in questo test le prove sono state svolte variando le features utilizzate, i risultati sono riportati di seguito.

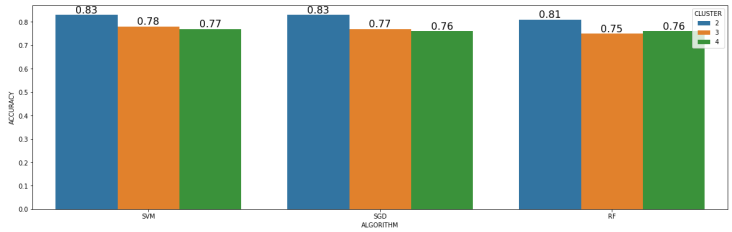

Figura 4.6: Accuracy con tutte le features

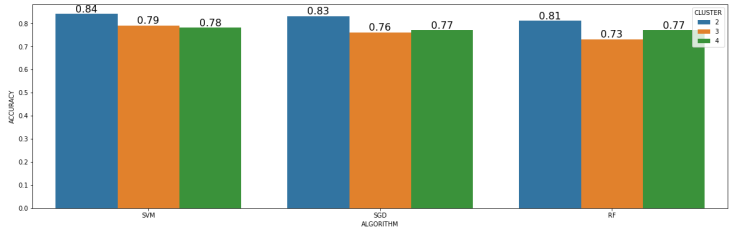

Figura 4.7: Accuracy con le migliori features

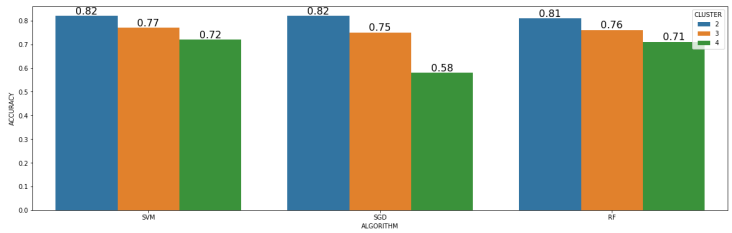

Figura 4.8: Accuracy con le peggiori features

Risultati

Come prevedibile, in seguito alle modifiche apportate al DataSet si può riscontrare un calo generale nell'accuratezza delle previsioni; il motivo è da ricercarsi nella precedente assegnazione del cluster predominante sul tratto di strada ai data points attestati sullo stesso tratto ma precedentemente assegnati a cluster differenti da quello prevalente, elemento questo che introduce un elemento di confusione in fase di training di un modello.

Per rispondere al problema del calo di accuratezza, in fase di testing è stato riprodotto sul test-set lo stesso meccanismo di assegnazione del cluster dominante sul segmento stradale, questa volta assegnando preventivamente come label i cluster prodotti dai classificatori sul medesimo test-set. Ultimata questa procedura, è stata ricontrollata l'accuracy. Contrariamente alle aspettative, questo esperimento ha comportato un ulteriore calo di accuratezza rendendo di fatto inutilizzabili i risultati ottenuti.

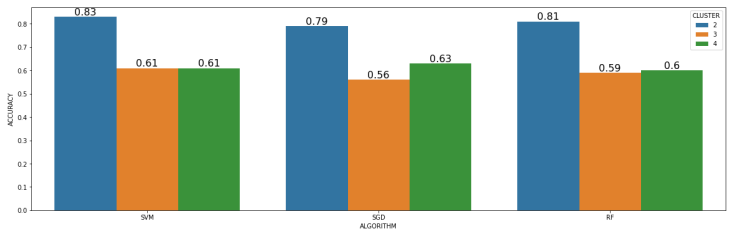

Figura 4.9: Accuracy con label modificate

Riscontro visuale

Per capire cosa effettivamente rappresentino i dati assegnati rispettivamente ai raggruppamenti a 3 e 4 cluster, è stato necessario ripassare dai punti segnalati dai data points che ne specificavano la posizione, al fine di prendere visione delle caratteristiche del manto stradale. Nella casistica in cui i segmenti di strada sono stati classificati utilizzando 3 cluster, un primo cluster identifica un terreno malmesso con ingente presenza di deformazioni dell'asfalto, un altro cluster identifica un tratto di strada senza particolari dissesti e l'ultimo cluster si attesta su una condizione del manto stradale

meno scadente rispetto a quella specificata dal primo cluster ma comunque non inquadrabile in quello che comunemente verrebbe definito un tratto di strada in buone condizioni. Nel caso a 4 cluster, come nel caso precedente, i primi due cluster identificano tratti di strada definibili in buone o cattive condizioni; i restanti due cluster si riferiscono a superfici stradali simili a quelle definite dal terzo cluster del caso precedente. Almeno in termini visivi, non è stato possibile distinguere una netta differenza tra questi due restanti cluster.

Conclusioni

Lo scopo di questa tesi era sapere se fosse possibile classificare differenti tipologie di manto stradale partendo dai dati dei sensori di movimento montati su uno smartphone. É stato appurato che con i dati raccolti con l'app appositamente sviluppata ed un algoritmo Supervised/Unsupervised è possibile raggiungere questo obiettivo. L'app RidingStyle memorizza correttamente i movimenti del ciclista mentre guida e tutti i valori rilevati dai sensori integrati sono registrati su un file csv. L'applicazione colleziona i dati in maniera tale che permette ad un algoritmo di Machine Learning di imparare a classificare le condizioni del manto stradale. Il migliore degli algoritmi testati è risultato essere SupportVectorMachine con un'accuratezza media dell'80% nel riconoscere i dati appartenenti ai raggruppamenti a 2 e 3 cluster rappresentanti differenti conformazioni di asfalto.

Bibliografia

[1] Oresti Banos, Juan-Manuel Galvez, Miguel Damas, Hector Pomares, and Ignacio Rojas. Window size impact in human activity recognition. *Sensors*, 14(4):6474–6499, 2014.

[2] Akanksh Basavaraju, Jing Du, Fujie Zhou, and Jim Ji. A machine learning approach to road surface anomaly assessment using smartphone sensors. *IEEE Sensors Journal*, 20(5):2635–2647, 2019.

[3] Sambit Kumar Beura, Haritha Chellapilla, and Prasanta Kumar Bhuyan. Urban road segment level of service based on bicycle users' perception under mixed traffic conditions. *Journal of modern transportation*, 25(2):90–105, 2017.

[4] Erhan Bulbul, Aydin Cetin, and Ibrahim Alper Dogru. Human activity recognition using smartphones. In *2018 2nd international symposium on multidisciplinary studies and innovative technologies (ismsit)*, pages 1–6. IEEE, 2018.

[5] Anna Ferrari, Daniela Micucci, Marco Mobilio, and Paolo Napoletano. Trends in human activity recognition using smartphones. *Journal of Reliable Intelligent Environments*, 7(3):189–213, 2021.

[6] Marius Hoffmann, Michael Mock, and Michael May. Road-quality classification and bump detection with bicycle-mounted smartphones. In *UDM@ IJCAI*, page 39. Citeseer, 2013.

[7] Jafet Morales and David Akopian. Physical activity recognition by smartphones, a survey. *Biocybernetics and Biomedical Engineering*, 37(3):388–400, 2017.

[8] Nitish Nag, Vaibhav Pandey, Aishwarya Manjunath, Avinash Vaka, and Ramesh Jain. Surface type estimation from gps tracked bicycle activities. *arXiv preprint arXiv:1809.09745*, 2018.

[9] Muhammad Shoaib, Stephan Bosch, Ozlem Durmaz Incel, Hans Scholten, and Paul JM Havinga. A survey of online activity recognition using mobile phones. *Sensors*, 15(1):2059–2085, 2015.

[10] Siying Zhu and Feng Zhu. Cycling comfort evaluation with instrumented probe bicycle. *Transportation research part A: policy and practice*, 129:217–231, 2019.

Ringraziamenti

Vorrei in primo luogo ringraziare la mia famiglia, ed in particolar modo i miei genitori, per il mai mancato sostegno morale ed economico negli anni di studi. Senza di voi non ce l'avrei fatta.
Vorrei ringraziare mia sorella Elisabetta e Marco semplicemente per quello che sono, due persone splendide. Un ringraziamento particolare va al mio relatore, Dott. Federico Montori, per la grande disponibilità e pazienza dimostrata; raramente ho incontrato un professore con la pazienza di spiegarmi la stessa cosa per tre volte in dieci minuti e proporsi per una quarta.
Vorrei infine ringraziare le persone con cui ho condiviso la mia quotidianità, con tutte le conseguenze del caso: i miei amici storici Antonio e Andrea, la fastidiosa Alessia, tutte le Raffe, Nedda e Federico, Anna e Massimiliano, Alice, Radu, Leo con cui abbiamo fin troppe cose in comune, Giovanni, Eleonora e chiunque altro mi sia stato di fianco in questi anni. Siete stati importanti, sappiatelo.