# Progetto Basket Shots

Coppola Matteo
793329

Palazzi Luca
793556

Vivace Antonio
793509

# Dominio e obiettivi, acquisizione

Obiettivo: predire l'esito di un tiro a canestro effettuato dai giocatori NBA a partire da informazioni sui giocatori e sulla configurazione di gioco.

- Individuazione di dataset complementari su cui effettuare l'integrazione
- Analisi delle misure di qualità, Data Preparation e Data Integration
- Metriche esplorative del dataset integrato
- Scelta di un modello opportuno di Apprendimento Automatico
- Esperimenti ed analisi delle performance
- Ottimizzazione del modello e considerazioni sul risultato

# Descrizione shot_logs

- 128000 istanze sui tiri a canestro effettuati nella stagione 2014-2015
- Considera 281 giocatori NBA
- Insieme di fattori che descrivono parzialmente il contesto in cui è avvenuto il tiro
- Esito registrato in shot_result
- Creazione del campo percentage_previous_game

# Descrizione season_stats

- Performance e statistiche degli atleti NBA
- Dal 1950 al 2017
- Presenza di indicatori complessi come PER
- Statistiche valide sia per attaccanti che per difensori

# Misure di qualità dei dataset

Season_stats:

- Mancata correttezza rispetto al modello, 2 attributi sono campi inutili derivati dalla fase di scraping
- Mancata completezza: lo 0,12% degli attributi per le istanze sono vuoti
- ID incrementale non opportuno

Shot_Logs:

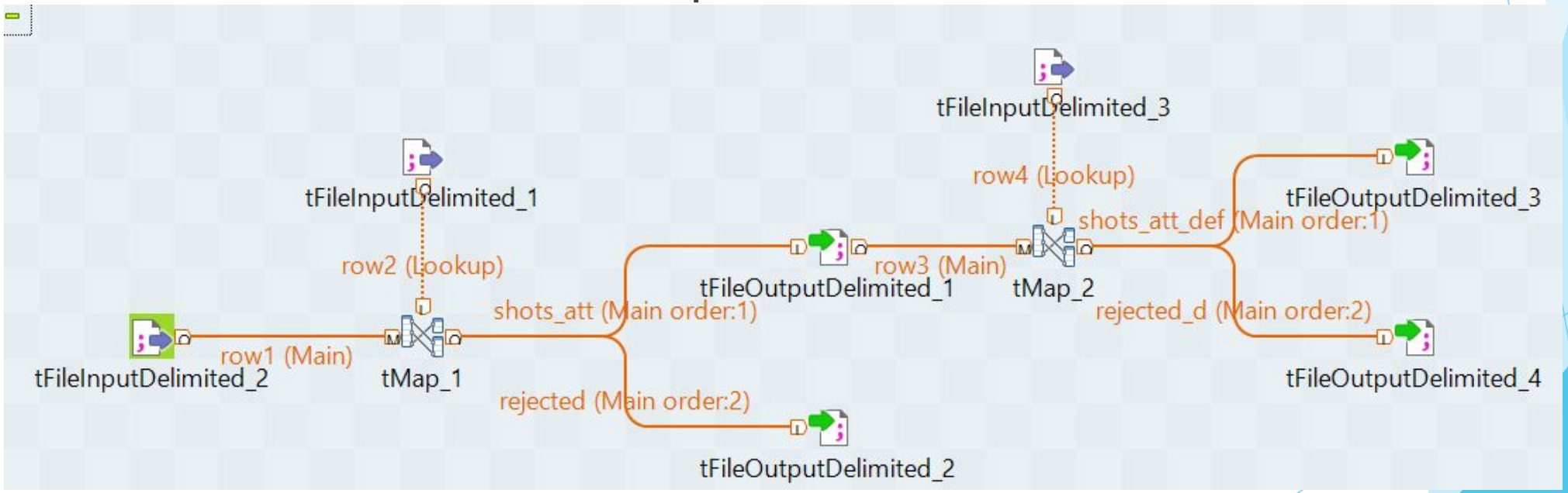
- 4,35% di incompletezza nei valori di shot_clock
- player_name e closest_def soffrono di eterogeneità interschema, stessi giocatori chiamati diversamente; ne soffrono tutte le istanze (100%)

# Data preparation

- Uniformare i nomi dei giocatori in player_name e closest_def
- Riempimento dei valori mancanti di shot_clock
- Selezione delle istanze in season_stats riferite ai giocatori che hanno giocato nel 2015
- Risoluzione (record linkage e data fusion) per le istanze multiple riferite ad uno stesso giocatore in season_stats
- Alcuni record rimossi perchè i giocatori non comparivano nell'altro dataset

# Data integration

- shot_logs -> attaccanti -> difensori
- Risoluzione dei match non perfetti

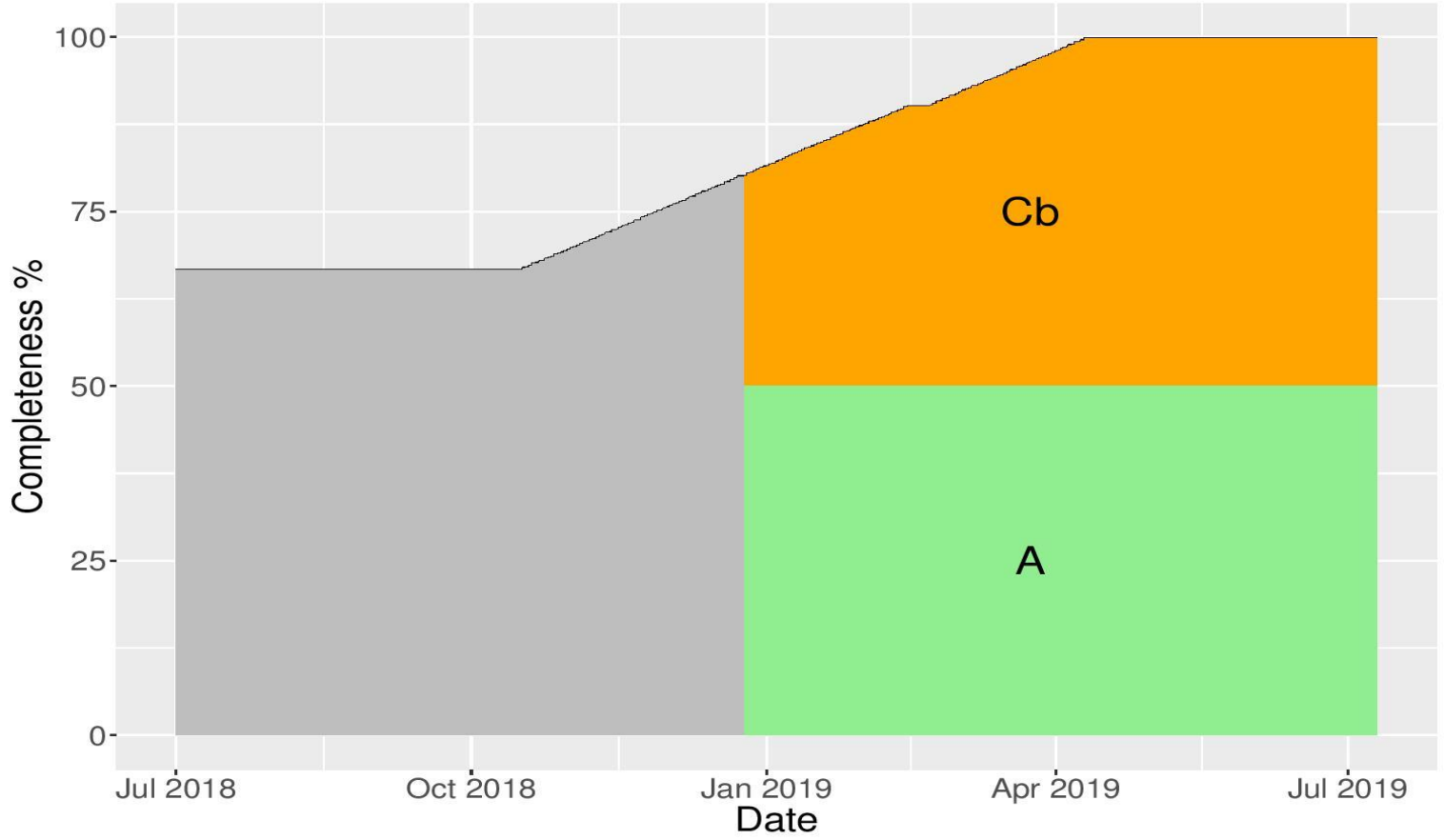


# Currency, Volatility e Timeliness

- Currency come velocità degli aggiornamenti
- Volatility come validità dei dati
- Timeliness come tempestività

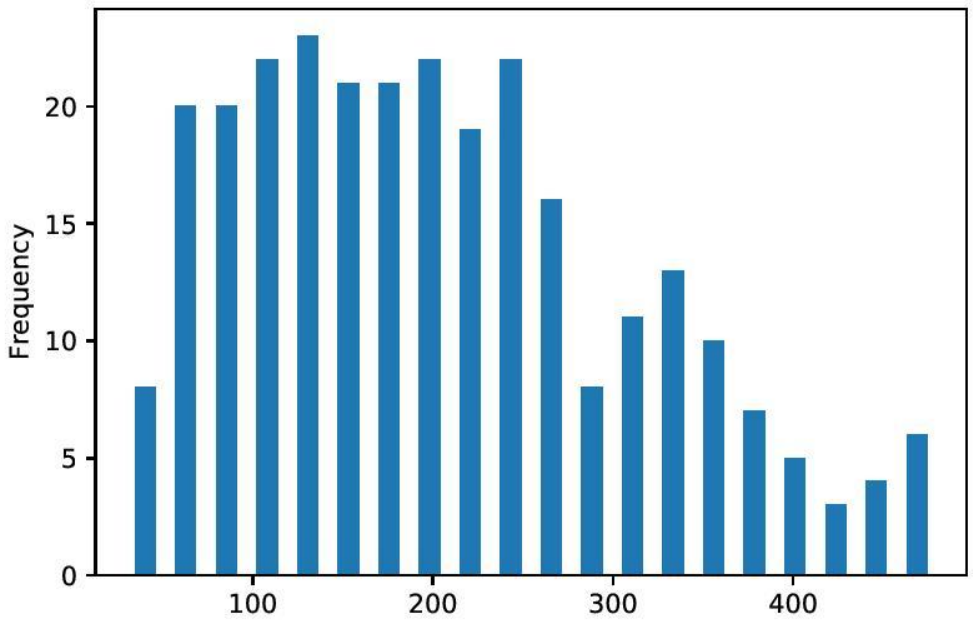
# Completability

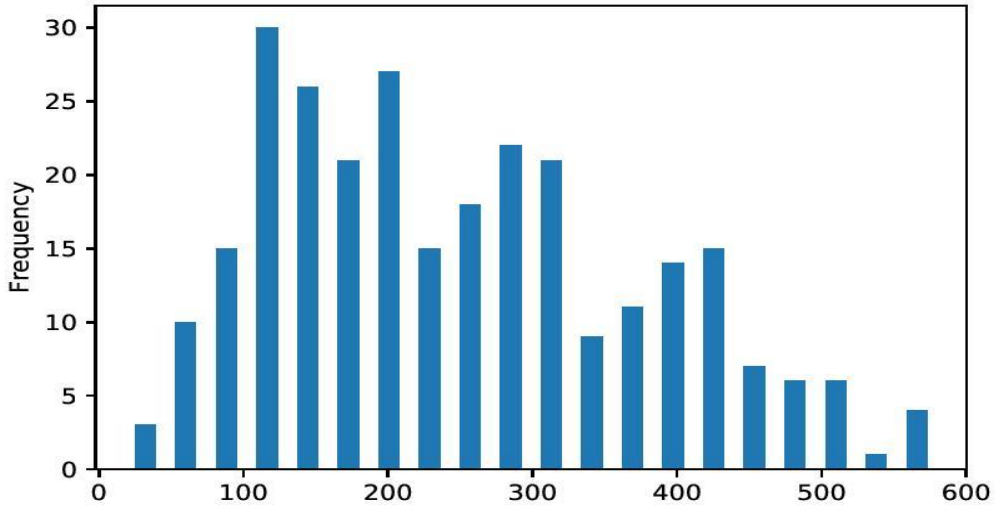
- Misura dell'evoluzione temporale della completezza
- Andamento periodico per la natura del campionato NBA



# Analisi descrittiva del dataset integrato

- Media di tiri in stagione: 455.72
- Media di blocchi; 271.31
- Media dei tiri con esito positivo; 206.05
- Media dei tiri con esito negativo; 249.67



# Dominio e obiettivi, descrizione dataset

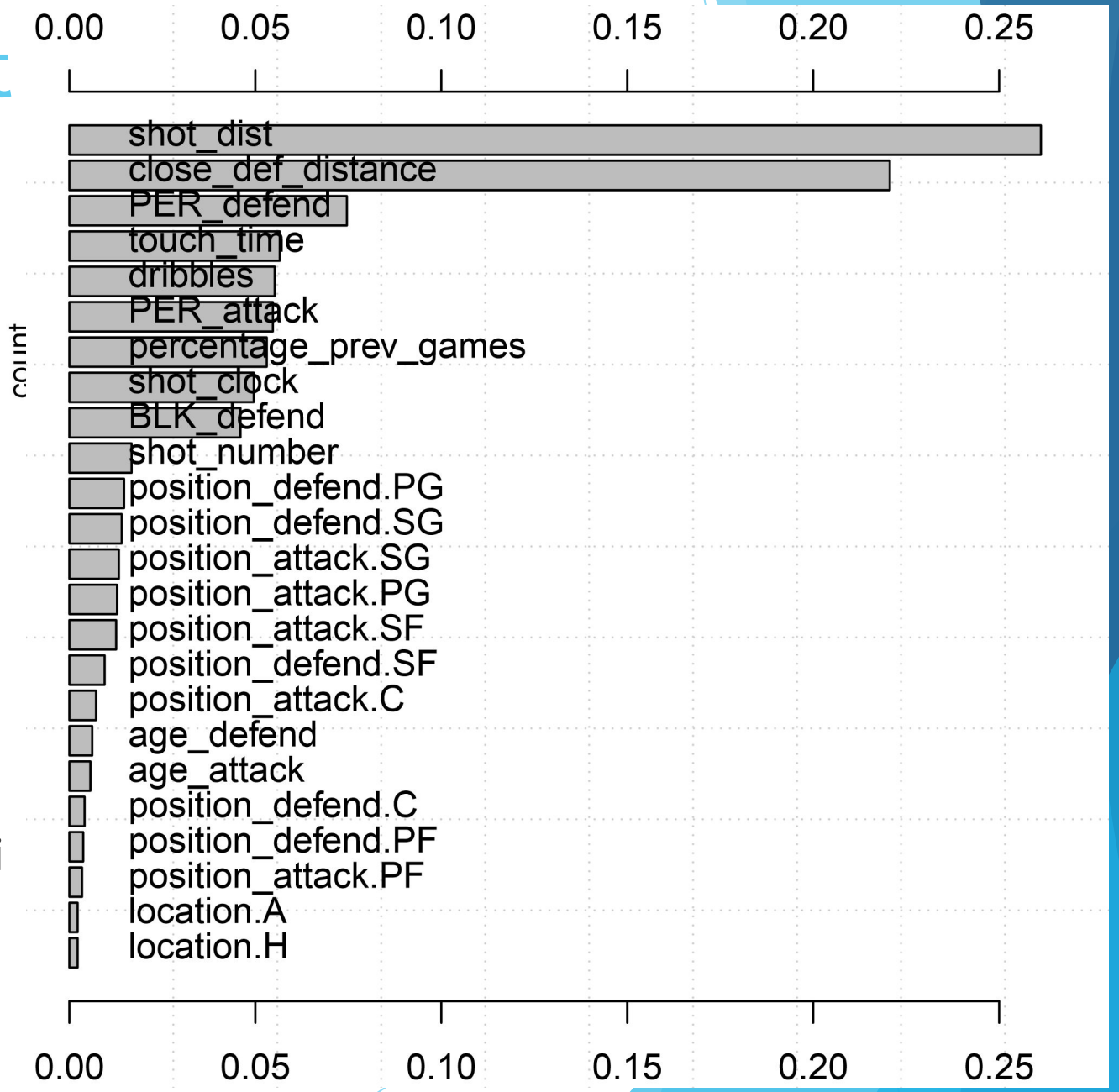
Obiettivo: predire l'esito di un tiro a canestro effettuato dai giocatori NBA a partire da informazioni sui giocatori e sulla configurazione di gioco.

- Individuazione di dataset complementari su cui effettuare l'integrazione
- Analisi delle misure di qualità, Data Preparation e Data Integration
- Metriche esplorative del dataset integrato
- Scelta di un modello opportuno di Apprendimento Automatico
- Esperimenti ed analisi delle performance
- Ottimizzazione del modello e considerazioni sul risultato

# Descrizione del dataset

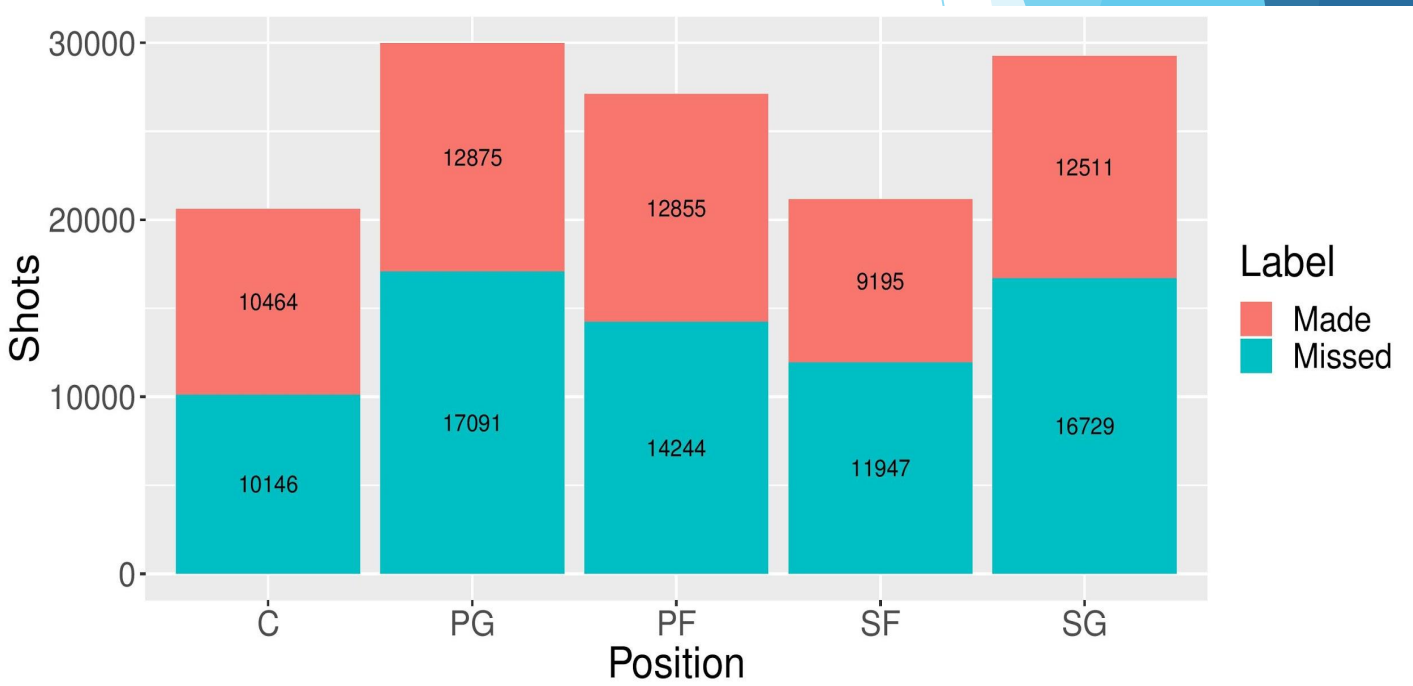
- 128 000 istanze su tiri a canestro effettuati nella stagione 2014-2015 abbinati alle informazioni su attaccanti e difensori coinvolti
- Attributi non rilevanti esclusi durante la fase di integrazione
- Problema di predizione binario: canestro made o missed

- Contributo informativo degli attributi



# Descrizione dataset

- Analisi esplorativa su due regioni
  - Asse delle ascisse [0, 10] e asse delle ordinate [5, 60]
  - Asse delle ascisse [25, 40] e asse delle ordinate [0, 20]
- Analisi della distanza media del tiro rispetto al ruolo del giocatore
- Analisi del successo dei tiri rispetto al ruolo del giocatore



# Modello di Machine Learning adottato

- SVM: algoritmo di apprendimento supervisionato che sfrutta il cosiddetto kernel-trick
- Scelto considerando il numero di osservazioni nel dataset e il numero di attributi selezionati
- La libreria e1071 aveva problemi di memoria
- La libreria liquidSVM non visualizzava le metriche
- SVM implementata con Rminer, libreria che implementa l'algoritmo di KernLabs e che permette il calcolo delle stime di probabilità
- Lanciando prima un'euristica su un subset più piccolo per trovare i valori di C e Kernel più opportuni

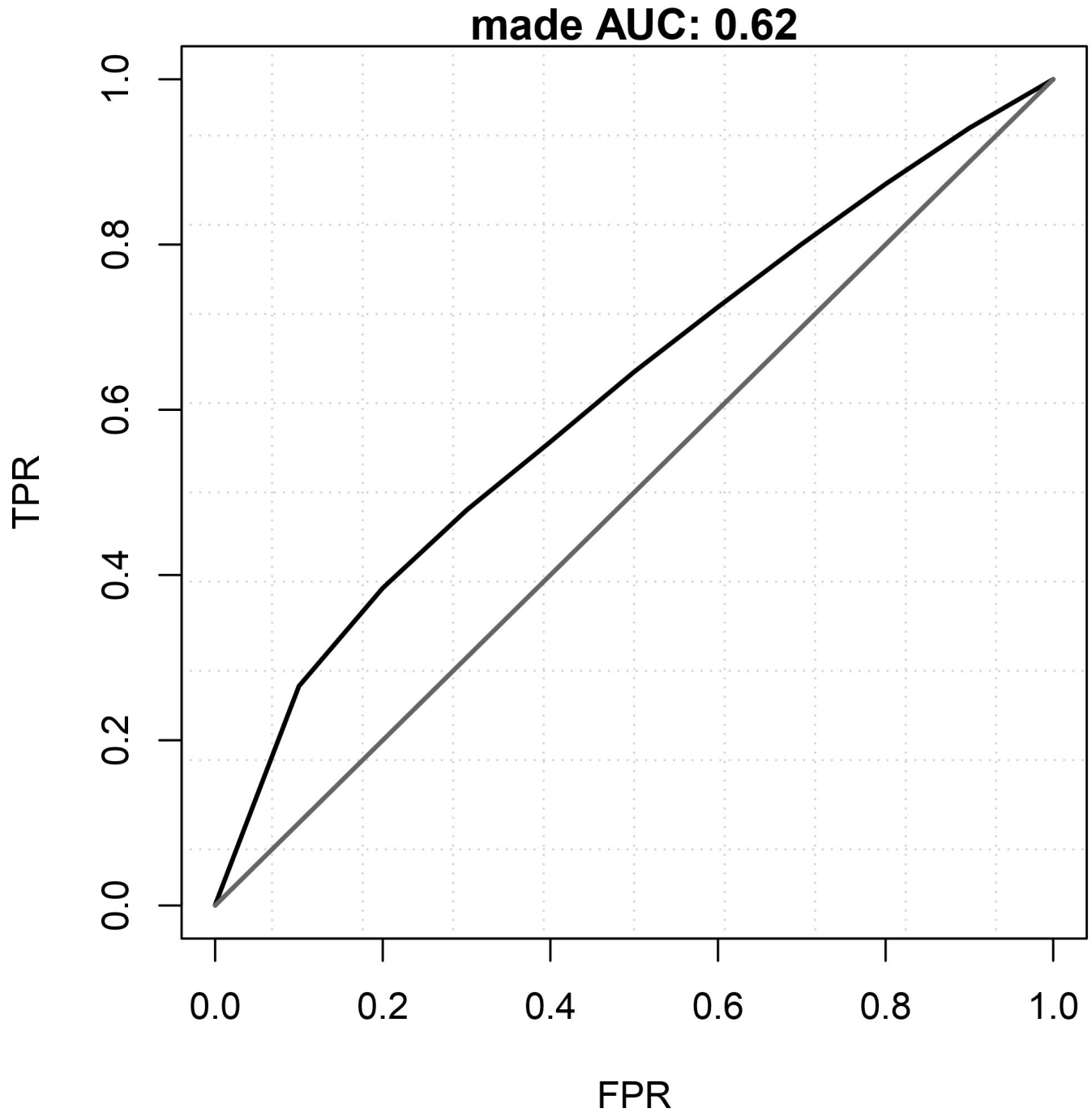
# Processo di creazione del dataset

- Pulizia del dataset svolto in gran parte in Talend nel passo di Data Integration
- Eliminati alcuni record con valore negativo per l'attributo touch time
- Normalizzazione degli attributi numerici utilizzando min max
- Tecnica di one hot encoding per gestire gli attributi categorici (incompatibili con il modello scelto)

# Esperimento ed analisi dei risultati

- Cross validation su 25 000 istanze con SVM
- Metriche:
  - Accuracy: 61.16
  - Precision per la classe made: 61.32
  - Precision per la classe missed: 61.09
  - Recall per la classe made: 38.49
  - Recall per la classe missed: 79.90
  - F-measure per la classe made: 47.29
  - F-measure per la classe missed: 69.24
- Area under curve per il problema di classificazione binario: 0.62

# AUROC



# Conclusioni

- I limiti vincolanti del problema
- Le metriche mancanti
- I risultati di Stanford
- Possibili sviluppi futuri