Università Politecnica delle Marche

Dipartimento di Ingegneria dell'Informazione

Corso di Laurea Magistrale in Ingegneria Informatica e dell'Automazione

Analisi sulle piogge in Australia e sui dati relativi alle gare di powerlifting con l'utilizzo di Python

Corso di Data Science

Professore

Prof. Domenico Ursino

Gruppo 1

Michele Pasqualini
Denil Nicolosi
Eris Prifti

Anno accademico 2021-2022

Indice

1 Dataset rain in Australia

Per effettuare analisi sulle serie temporali, è necessario identificare un dataset idoneo con dati che variano nel tempo. La nostra scelta è ricaduta sul dataset "Rain in Australia" disponibile su Kaggle al seguente link kaggle.com/datasets/jsphyg/weather-dataset-rattle-package), un dataset che contiene circa 10 anni di osservazioni meteorologiche giornaliere da molte località in tutta l'Australia, eseguite dal 1 novembre 2007 al 25 giugno 2017.

1.1 Struttura del dataset

Il dataset in questione è composto da diversi campi, essi vengono mostrati nella seguente tabella:

Date	La data di osservazione.
Location	Il nome della posizione della stazione meteorologica.
MinTemp	La temperatura minima in gradi centigradi.
Maxtemp	La temperatura massima in gradi centigradi.
Rainfall	La quantità di precipitazioni registrate per la giornata in mm.
Evaporation	Quantità di evaporazione in mm nelle 24 ore.
Sunshine	Il numero di ore di sole nel corso della giornata.
WindGustDir	La direzione della raffica di vento più forte nelle 24 ore.
WindGustSpeed	La velocità (km/h) della raffica di vento più forte nelle 24
WindDir9am	Direzione del vento alle 9:00.
WindDir3pm	Direzione del vento alle 15:00.
WindSpeed9am	La velocità del vento (km/h) media alle 9:00.
WindSpeed3pm	La velocità del vento (km/h) media alle 15:00.
Humidity9am	Umidità (percentuale) alle 9:00.
Humidity3pm	Umidità (percentuale) alle 15:00.
Pressure9am	La pressione atmosferica (hpa) alle 9:00.
Pressure3pm	La pressione atmosferica (hpa) alle 15:00.
Cloud9am	Registra quanti ottavi del cielo sono oscurati dalle nuvole alle 9:00.
Cloud3pm	Registra quanti ottavi del cielo sono oscurati dalle nuvole alle 15:00.
Temp9am	Temperatura (gradi) alle 9:00.
Temp3pm	Temperatura (gradi) alle 15:00.
RainToday	Booleano: 1 se la precipitazione (mm) nelle 24 supera 1 mm, altrimenti 0.
RainTomorrow	La quantità di pioggia del giorno successivo in mm. Utilizzato per creare la variabile di risposta RainTomorrow. Una sorta di misura del "rischio".

Tabella 1: Descrizione campi del dataset

Successivamente viene svolta un'analisi del dataset per identificare quali campi contengono valori "Nan" o non contengono a valori. Attraverso il comando `weather.isnull().sum()`, illustrato in figura 1 è stato possibile visualizzare la quantità di valori mancanti in ogni campo del dataset. A fronte di tutto questo, però, abbiamo ritenuto non necessario eliminare tali valori, perchè potevano essere eliminate alcune righe del dataset che contenevano informazioni utili ai fini dell'analisi delle serie temporali.

```
Date                0
Location            0
MinTemp          1485
MaxTemp          1261
Rainfall         3261
Evaporation     62790
Sunshine        69835
WindGustDir     10326
WindGustSpeed   10263
WindDir9am      10566
WindDir3pm       4228
WindSpeed9am     1767
WindSpeed3pm     3062
Humidity9am      2654
Humidity3pm      4507
Pressure9am     15065
Pressure3pm     15028
Cloud9am        55888
Cloud3pm        59358
Temp9am          1767
Temp3pm          3609
RainToday        3261
RainTomorrow     3267
dtype: int64
```

Figura 1: Output del comando `weather.isnull().sum()`

Attraverso il comando `weather.describe()` si ottiene in output la tabella in figura 2 in cui si visualizzano delle statistiche descrittive che riassumono la tendenza centrale, la dispersione e la forma della distribuzione di un set di dati, escludendo i valori NaN.

	MinTemp	MaxTemp	Rainfall	Evaporation	Sunshine	WindGustSpeed	WindSpeed9am	WindSpeed3pm
count	143.975.000.000	144.199.000.000	142.199.000.000	82.670.000.000	75.625.000.000	135.197.000.000	143.693.000.000	142.398.000.000
mean	12.194.034	23.221.348	2.360.918	5.468.232	7.611.178	40.035.230	14.043.426	18.662.657
std	6.398.495	7.119.049	8.478.060	4.193.704	3.785.483	13.607.062	8.915.375	8.809.800
min	-8.500.000	-4.800.000	0.000000	0.000000	0.000000	6.000.000	0.000000	0.000000
25%	7.600.000	17.900.000	0.000000	2.600.000	4.800.000	31.000.000	7.000.000	13.000.000
50%	12.000.000	22.600.000	0.000000	4.800.000	8.400.000	39.000.000	13.000.000	19.000.000
75%	16.900.000	28.200.000	0.800000	7.400.000	10.600.000	48.000.000	19.000.000	24.000.000
max	33.900.000	48.100.000	371.000.000	145.000.000	14.500.000	135.000.000	130.000.000	87.000.000

	Humidity9am	Humidity3pm	Pressure9am	Pressure3pm	Cloud9am	Cloud3pm	Temp9am	Temp3pm
count	142.806.000.000	140.953.000.000	13.039.500.000	130.432.000.000	89.572.000.000	86.102.000.000	143.693.000.000	14.185.100.000
mean	68.880.831	51.539.116	101.764.994	1.015.255.889	4.447.461	4.509.930	16.990.631	2.168.339
std	19.029.164	20.795.902	710.653	7.037.414	2.887.159	2.720.357	6.488.753	693.665
min	0.000000	0.000000	98.050.000	977.100.000	0.000000	0.000000	-7.200.000	-540.000
25%	57.000.000	37.000.000	101.290.000	1.010.400.000	1.000.000	2.000.000	12.300.000	1.660.000
50%	70.000.000	52.000.000	101.760.000	1.015.200.000	5.000.000	5.000.000	16.700.000	2.110.000
75%	83.000.000	66.000.000	102.240.000	1.020.000.000	7.000.000	7.000.000	21.600.000	2.640.000
max	100.000.000	100.000.000	104.100.000	1.039.600.000	9.000.000	9.000.000	40.200.000	4.670.000

Figura 2: Output del comando `weather.describe()`

1.2 Visualizzazione dei dati

1.2.1 Distribuzione dei dati suddivisi per anni

In questo grafico si può visualizzare la distribuzione dei dati nei vari anni. Come già riportato, i dati raccolti nel dataset vanno dal 1 novembre 2007 al 25 giugno 2017. Proprio per il fatto che i dati raccolti nel 2017 terminano a giugno, si può vedere dalla figura che essi sono circa la metà rispetto a quelli degli altri anni. Invece nel 2007 sono presenti pochi dati, questo perchè nel dataset si trovano solamente due mesi riferiti a quell'anno e la raccolta dati si è intensificata a partire dall'anno 2009. Negli anni successivi, invece, il regime di campionamento è più o meno costante, mantenendo una media di circa 16000 campioni all'anno.

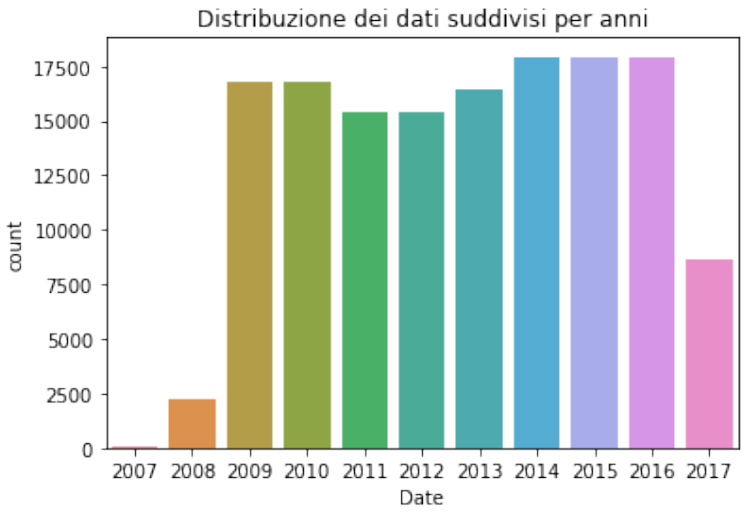


Figura 3: Distribuzione dei dati suddivisi per anni

1.2.2 Relazioni

Nel grafico in figura 4 viene visualizzato il rapporto tra i campi Evaporation e Sunshine. Si può notare che fino a 8 ore di sole, la quantità di acqua evaporata si aggira sempre sullo stesso livello, mentre dalle 10 ore in poi l'evaporazione aumenta notevolmente fino a raddoppiare. Inoltre, aumenta anche la variabilità.

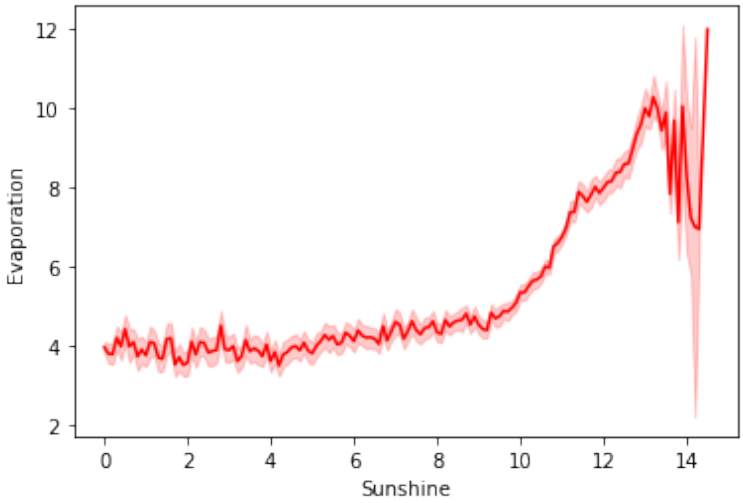


Figura 4: Relationship Sunshine-Evaporation

Nel grafico in figura 5 viene rappresentata la relazione che sussiste tra i campi Rainfall e Sunshine. In particolare, si può notare che l'andamento è decrescente. Infatti all'aumentare della scala dei valori di Sunshine, il valore di Rainfall diminuisce. Questa correlazione tra i dati può essere interpretata con il fenomeno che accade quando all'aumentare delle ore del sole, le piogge non si verificano.

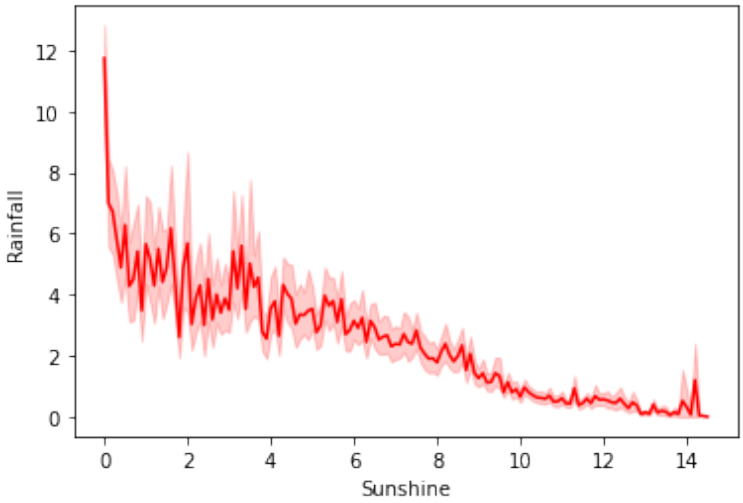


Figura 5: Relationship Sunshine-Rainfall

Nella visualizzazione 6 andiamo ad analizzare l'andamento dei valori delle temperature massime e minime rispetto al campo Sunshine. Si può notare che quando abbiamo un valore persistente di Sunshine, i valori delle

temperature tendono a crescere ed innalzarsi, per poi avere un leggera decrescita. Questo aspetto è dovuto alla variabilità che abbiamo nei dati, ciò significa che nel dataset potremmo avere certi intervalli temporali in cui le ore di Sunshine sono svariate, ma le temperatura restano comunque basse, oppure viceversa. Questo, ad esempio, potrebbe verificarsi in alcuni mesi come dicembre, gennaio, etc.

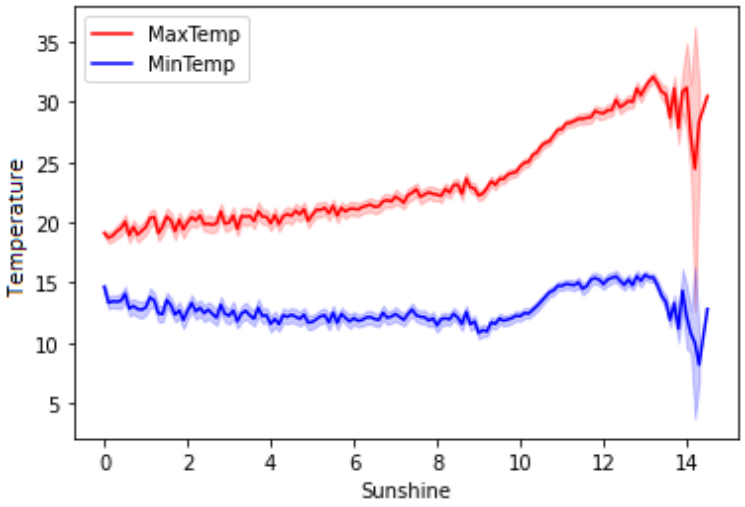


Figura 6: Relationship Sunshine-MinTemp & MaxTemp

1.2.3 Città con il conteggio su RainToday

In questo grafico viene riportato il conteggio del campo RainToday relativo ad ogni città. Ovvero, per ogni città australiana, andiamo a vedere se in un preciso giorno ha piovuto oppure no. Come si può osservare, su ogni città sono molto più numerose le volte in cui non cade pioggia.

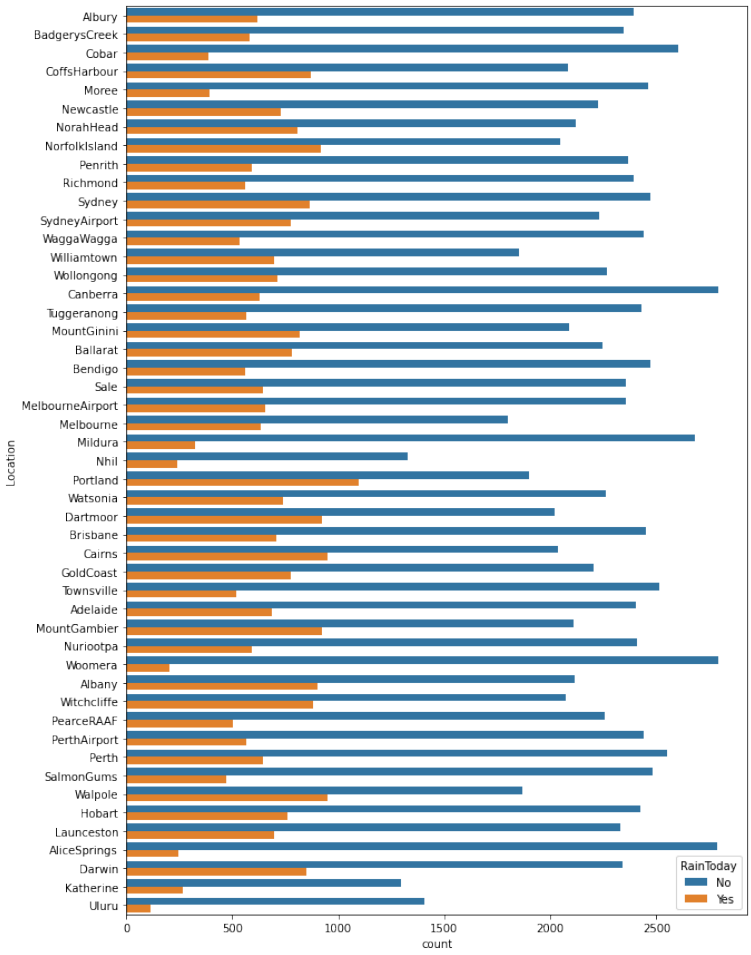


Figura 7: Conteggio di Raintoday per ogni città

1.2.4 Top 20 città in cui è caduta più pioggia

Nella figura 8 mostriamo una classifica delle 20 città in cui è caduta la maggior quantità di pioggia nell'intervallo di tempo che va dal 2007 al 2017. Si può notare dal grafico che la città con in cui è caduta la maggiore quantità di pioggia è Coffsharbour, seguita poi da Darwin, Cairns e Newcastle.

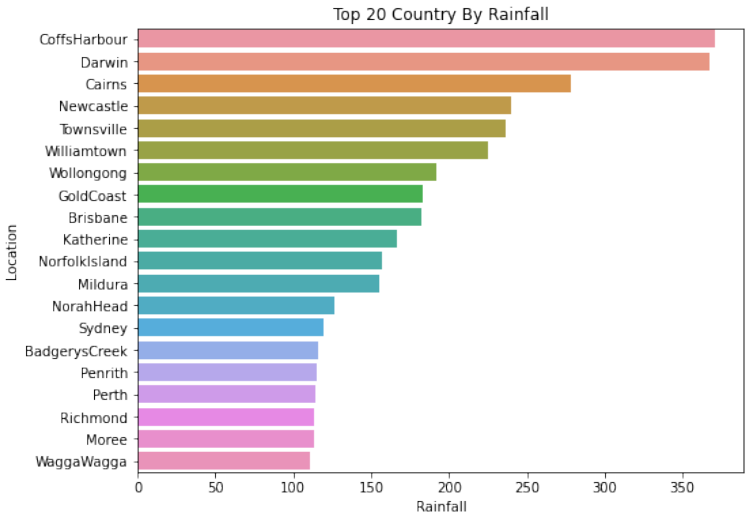


Figura 8: Top 20 città in cui è caduta più pioggia

1.2.5 Pioggia caduta ogni anno

In questo paragrafo si va ad illustrare la quantità di pioggia caduta in Australia per ogni anno del dataset. La quantità di pioggia per anno è stata calcolata sommando tutte le quantità di pioggia che si sono verificate in quell'anno. Come si può vedere dalla figura 9, gli anni in cui ha piovuto maggiormente sono stati gli anni 2009 e 2011.

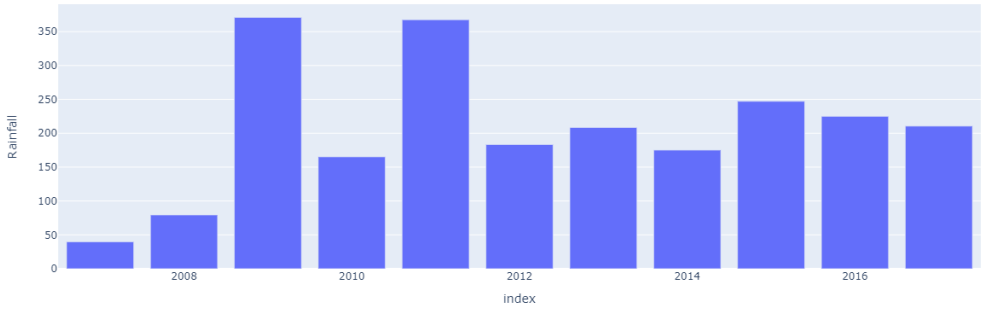


Figura 9: Anni in cui è caduta più pioggia

1.2.6 Correlazione tra i dati

In questo paragrafo viene testato il rapporto di correlazione tra i dati, al fine di trovare dei collegamenti utili tra i vari campi. Da questo emerge che:

- tra il campo "MinTemp" e "Temp9am" c'è un alto tasso di correlazione (0.9) e ciò può indicare che la temperatura che si registra alle 9 della mattina è sempre molto vicina alla temperatura minima della giornata.
- anche tra il campo "MaxTemp" e "Temp3pm" c'è un alto tasso di correlazione (0.98) e ciò può indicare che la temperatura che si registra alle 3 del pomeriggio è molto vicina ai valori della temperatura massima della giornata.

- tra i campi "Pressure9am" e "Pressure3pm" vi è una forte correlazione (0.96) e questo indica che la pressione nei due orari è molto simile.

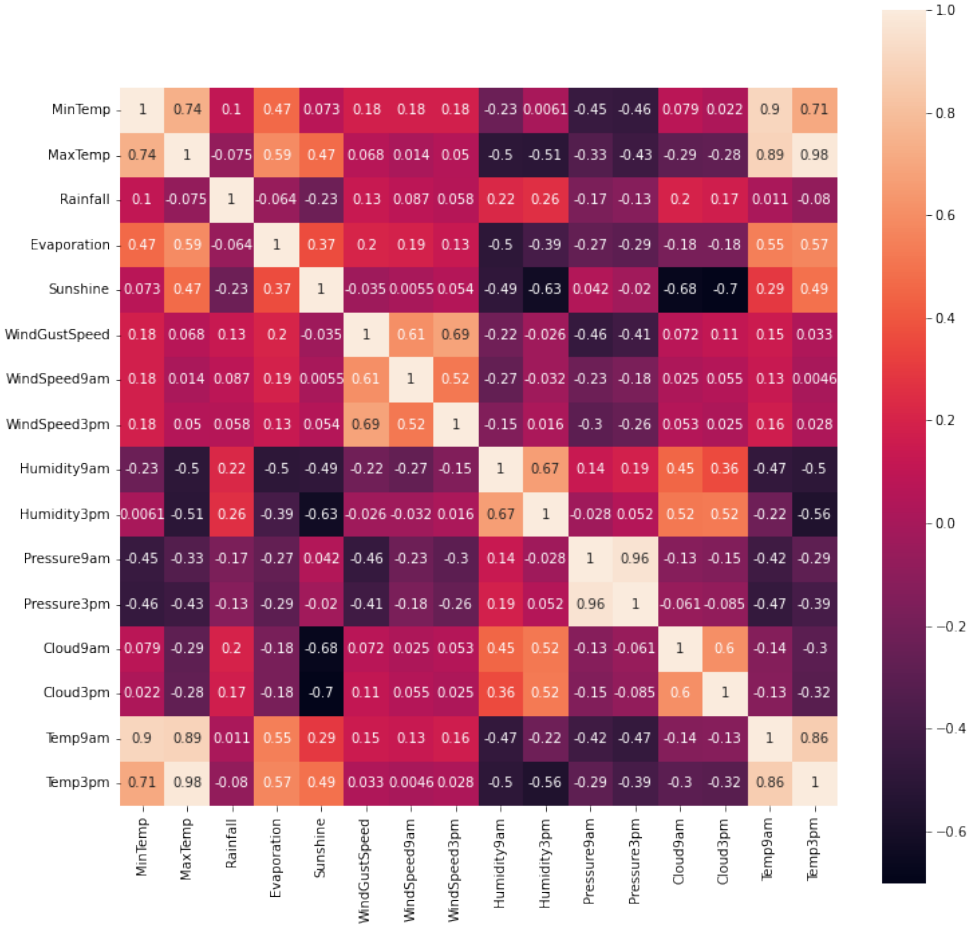


Figura 10: Correlazioni tra i dati

Una volta avuta una prima analisi di correlazione numerica, abbiamo effettuato anche una analisi grafica, riportando un grafico per ogni coppia di valori interessanti. Grazie alla figura 11 possiamo confermare le considerazioni precedenti. In più, si è possibile dire che:

- il campo "Rainfall" è condizionato molto da tutti gli altri campi. Infatti, con i campi riguardati la temperatura, umidità e pressione, si nota che ci sono diversi picchi nel grafico il quale simboleggiano che con certi valori la pioggia è molto più probabile e abbondante.
- la stessa cosa si può dire per "Evaporation", il quale si nota essere delimitata soprattutto dalla pressione.
- la pressione e la temperatura, nonostante non abbiano una stretta correlazione o un andamento, risultano essere precisamente correlati entro una certa area del grafico.

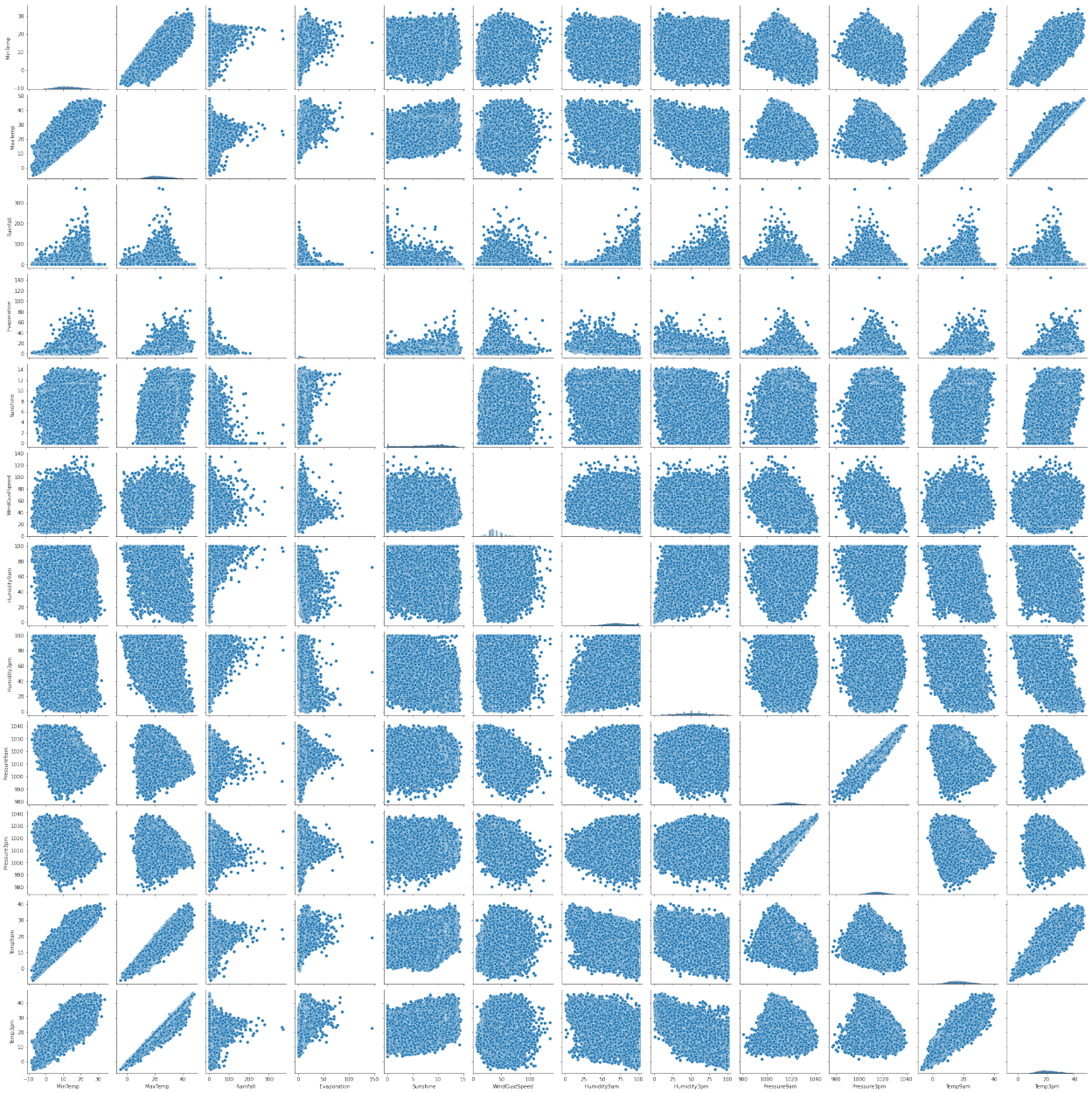


Figura 11: Correlazioni tra i dati

1.3 Serie temporali

In questa sezione viene descritto il procedimento utilizzato per l'analisi delle serie temporali. Come prima cosa, è stato creato una sorta di filtro il quale ci permette di selezionare l'intervallo temporale e la città su cui analizzare la serie. Ciò su cui verrà effettuata la previsione è la temperatura minima nella città di Canberra dal 2007 al 2020. I dati a disposizione sono giornalieri, ma per ridurre la quantità eccessiva dei dati sono stati raggruppati per mesi facendo una media.

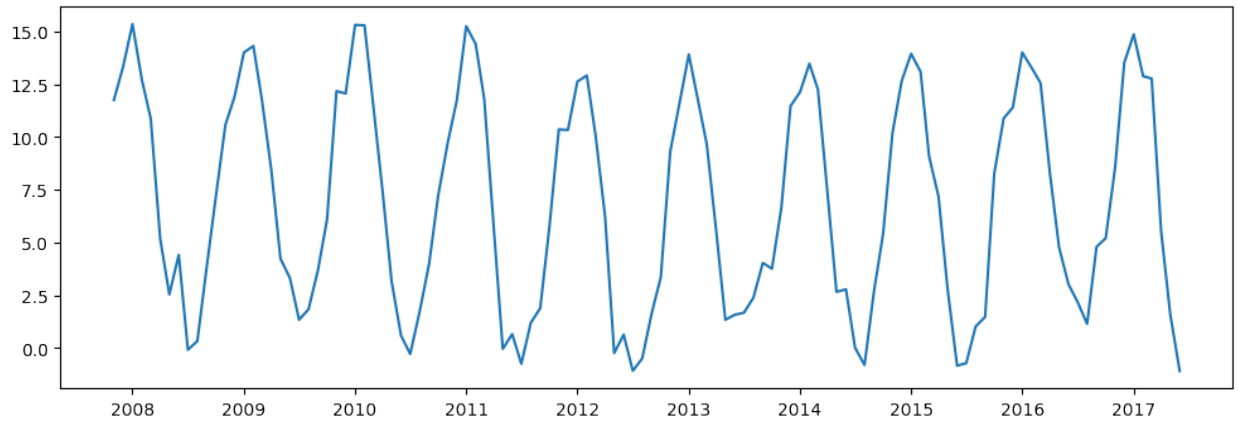


Figura 12: Andamento serie temporale su MinTemp

Dalla figura 14 si può notare come la serie abbia un andamento abbastanza uniforme lungo l'asse delle ascisse, potremmo ipotizzare che essa sia stazionaria. Ovviamente, quanto affermato non basta per classificare la serie come stazionaria, ma è necessario effettuare diversi test per dedurre se essa è effettivamente stazionaria o meno. Il primo test utilizzato è l'ADFuller il quale restituisce un p-value con valore pari a 0.154494. In questo test l'ipotesi nulla da verificare è che la serie non è stazionaria, l'ipotesi alternativa è che la serie è stazionaria. Dato che il p-value ottenuto è maggiore di 0.05 l'ipotesi nulla è verificata. Il secondo test che abbiamo deciso di effettuare è il KPSS test il quale restituisce un p-value con valore pari ad 0.100000. In questo test l'ipotesi nulla da verificare è che la serie è stazionaria, l'ipotesi alternativa è che la serie è non stazionaria. Dato che il p-value ottenuto è maggiore di 0.05, l'ipotesi nulla è verificata. Quindi, dal primo test risulta che la serie è non stazionaria, mentre dal secondo abbiamo che la serie è stazionaria. Tra i due metodi, quello che prevale è l'ADFuller e ciò ci permette affermare che la serie temporale in questione è non stazionaria.

```
ADF Statistic: -2.356160
p-value: 0.154494
```

```
Results of KPSS Test:
Test Statistic            0.207032
p-value                   0.100000
#Lags Used               13.000000
Critical Value (10%)      0.347000
Critical Value (5%)       0.463000
Critical Value (2.5%)     0.574000
Critical Value (1%)       0.739000
```

Figura 13: ADFuller e KPSS test

Poiché la serie non è stazionaria, facciamo una differenziazione della serie e vediamo come si presenta il grafico di autocorrelazione.

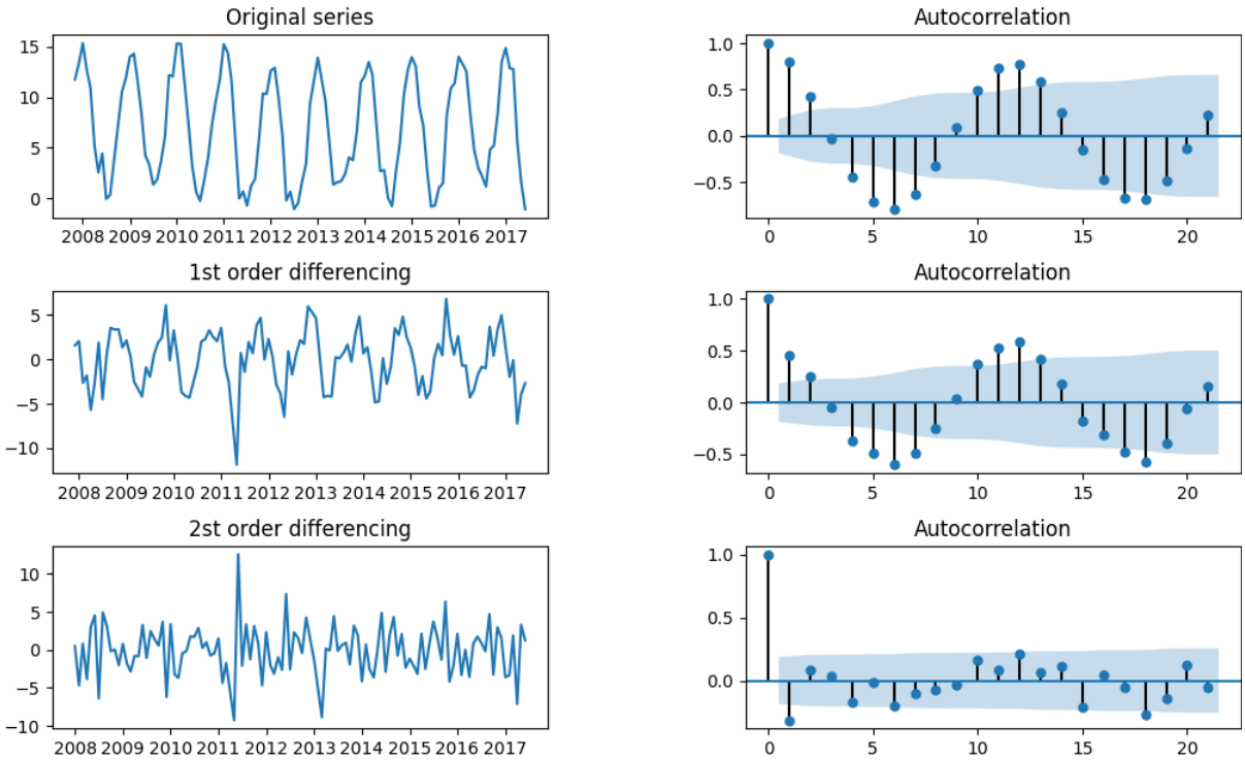


Figura 14: Differenziazione della serie temporale

Dall'immagine 14 è possibile osservare la serie temporale originale affiancata al grafico sulle autocorrelazioni. Di quest'ultimo vengono analizzati i lag che esso presenta, ovvero tutti quei valori che escono fuori dall'area di ammissibilità (area celeste). Nel nostro caso esistono diversi lag che non rientrano nella superficie raffigurata, ciò descrive il fatto che la serie potrebbe essere stazionaria. Andiamo a differenziare di un ordine successivo. La seconda riga dell'immagine 14 mostra il risultato della differenziazione di primo ordine. E' possibile ancora vedere come alcuni lag escono fuori dall'area, quindi è necessario differenziare ancora di un ordine. Nella parte finale dell'immagine vediamo l'andamento della serie con due ordini di differenza. Possiamo dire che la serie temporale raggiunge la stazionarietà con due ordini di differenza. Però, osservando l'autocorrelazione del primo grafico, emerge che la serie è stazionaria già nella serie originale (perché diversi lag escono dall'area di ammissibilità). Nell'autocorrelazione di secondo ordine di differenza, i lag entrano nella zona negativa abbastanza velocemente, il che indica che la serie potrebbe essere stata differenziata troppo, infatti vediamo come essa abbia un andamento randomico.

Inoltre, avendo diversi lag che escono dall'area di ammissibilità nel primo ordine di differenziazione, abbiamo ritenuto opportuno rieseguire il test ADFuller sulla serie al primo ordine di differenza. Il risultato ottenuto dal test ha restituito un p-value con valore pari a zero. Questo ci permette di affermare con sicurezza che la serie temporale che stiamo analizzando è stazionaria al primo ordine di differenziazione. Dato che inizialmente dall'ADFuller test risulta che la serie è non stazionaria, non possiamo impostare il parametro d uguale a zero. Invece, eseguendo l'ADFuller test al primo ordine di differenza risulta che la serie è stazionaria. Questo ci permette di assegnare al parametro d il valore 1.
Il passo successivo è quello di identificare se il modello ha bisogno di un contributo da parte della componente AR. È possibile scoprire l'ordine di AR necessari ispezionando il grafico dell'Autocorrelazione Parziale (PACF).

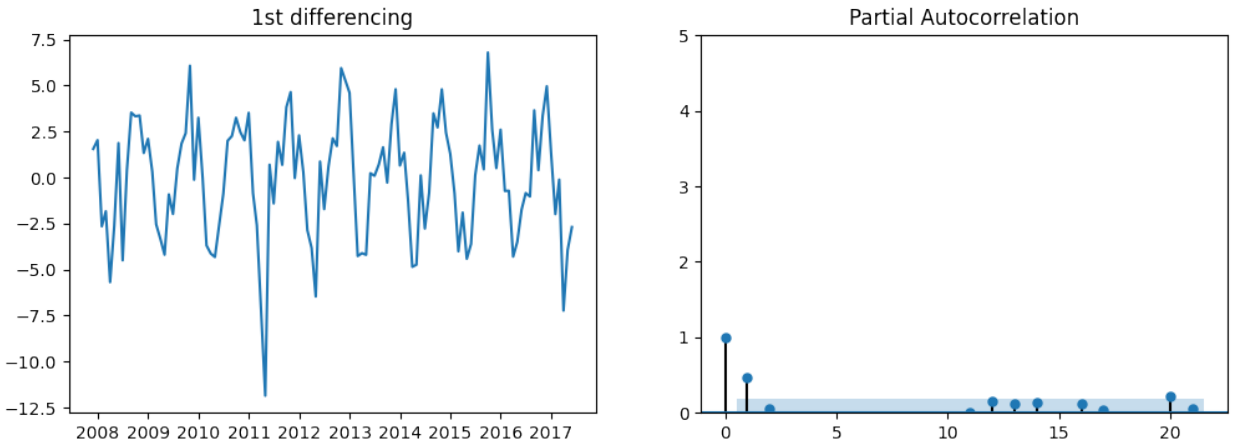


Figura 15: Autocorrelazione parziale della serie

Si può osservare che i lag significativi il quale si trovano al di sopra dell'area di ammissibilità sono due. Quindi, questo ci permette, per il momento, di fissare il valore di p pari a 2.

Nello stesso modo in cui è stato analizzato il grafico PACF per il numero di termini AR, si può guardare il grafico ACF per il numero di termini MA. L'ACF indica quanti termini MA sono necessari per rimuovere qualsiasi autocorrelazione nella serie stazionaria.

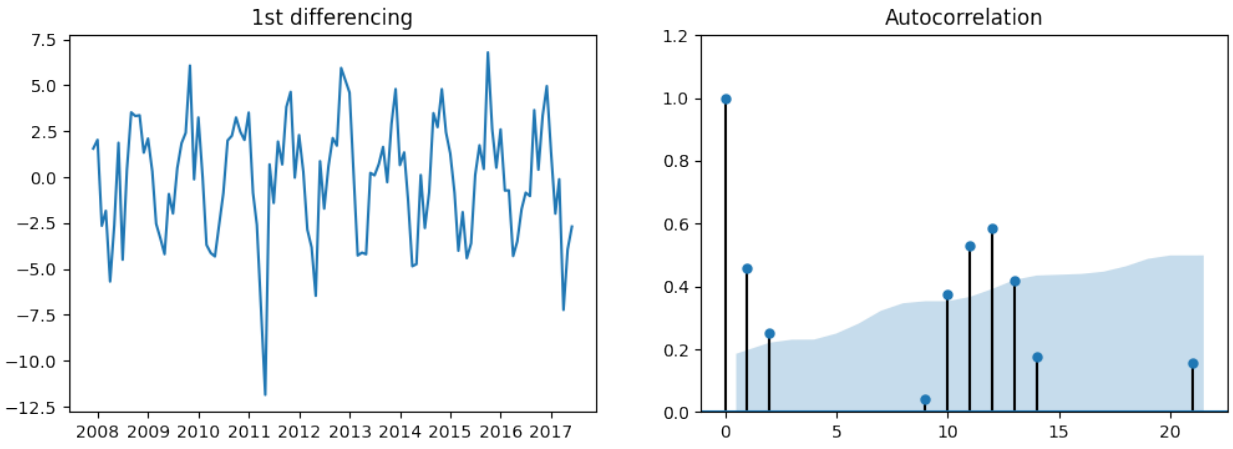


Figura 16: Autocorrelazione della serie

Dall'immagine 16 si nota che un paio di ritardi sono ben al di sopra della linea di significato. Quindi, viene fissato provvisoriamente il parametro q uguale a 2. In questo modo abbiamo determinato tutti i parametri per allenare il modello ARIMA, ovvero p=2, d=1, q=2.

ARIMA Model Results

Dep. Variable:	D.MinTemp	No. Observations:	112
Model:	ARIMA(2, 1, 2)	Log Likelihood	-234.145
Method:	css-mle	S.D. of innovations	1.912
Date:	Wed, 13 Apr 2022	AIC	480.289
Time:	14:50:14	BIC	496.600
Sample:	1	HQIC	486.907

	coef	std err	z	P>|z|	[0.025	0.975]
const	-0.0171	0.071	-0.240	0.811	-0.157	0.122
ar.L1.D.MinTemp	1.6998	0.020	85.030	0.000	1.661	1.739
ar.L2.D.MinTemp	-0.9732	0.018	-52.912	0.000	-1.009	-0.937
ma.L1.D.MinTemp	-1.7775	0.047	-37.625	0.000	-1.870	-1.685
ma.L2.D.MinTemp	0.8842	0.047	18.729	0.000	0.792	0.977

Roots

	Real	Imaginary	Modulus	Frequency
AR.1	0.8733	-0.5147j	1.0137	-0.0848
AR.2	0.8733	+0.5147j	1.0137	0.0848
MA.1	1.0052	-0.3472j	1.0635	-0.0529
MA.2	1.0052	+0.3472j	1.0635	0.0529

Figura 17: Valori del modello ARIMA

Si noti che tutti i valori nella colonna P> |z| sono molto bassi (minori di 0.05), quindi il modello è significativo. Si noti che anche il valore MA2 è a 0 quindi il valore di p non ha bisogno di essere decrementato.

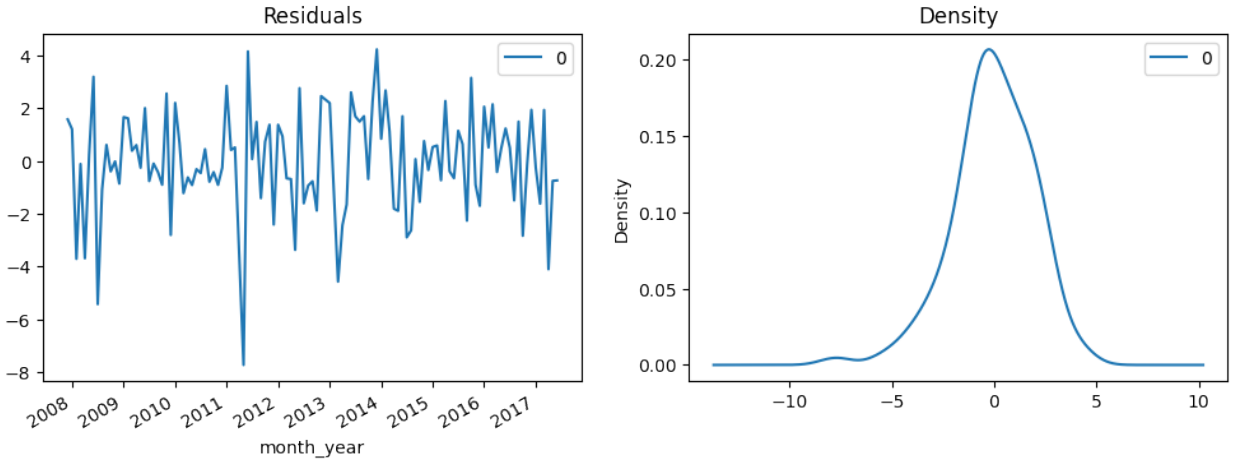


Figura 18: Residui della serie temporale

Quindi viene realizzato un grafico dei residui per osservare l'eventuale presenza di pattern particolari. Si cerca quindi una varianza e media costante (i valori sono compresi tra [-5,5] e con una media prossima allo 0)

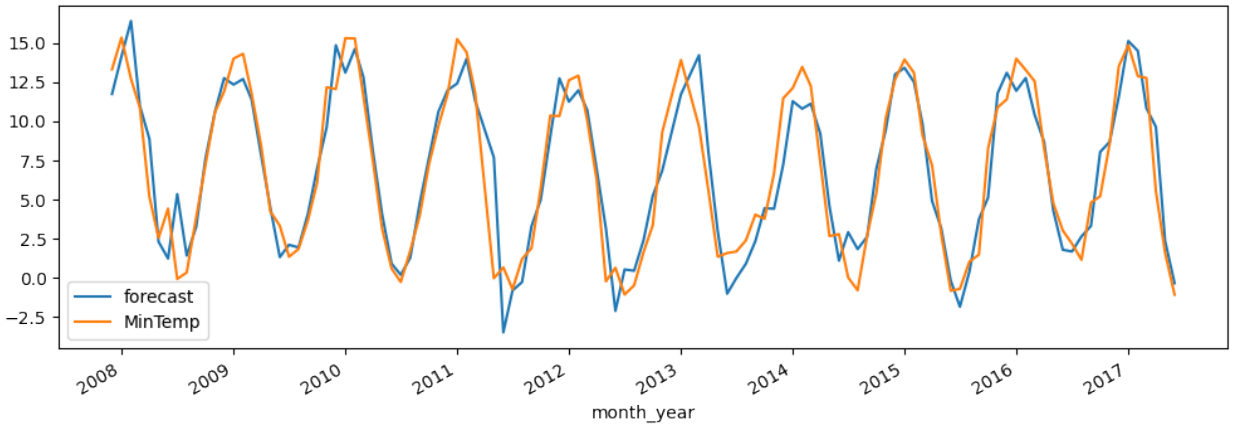


Figura 19: Predizione serie temporale

Una volta allenato il modello ARIMA con i parametri sopra descritti, nella figura 19 viene mostrata la predizione sui dati presi in considerazione. In questo caso, i valori in-sample lagged vengono utilizzati per la previsione.

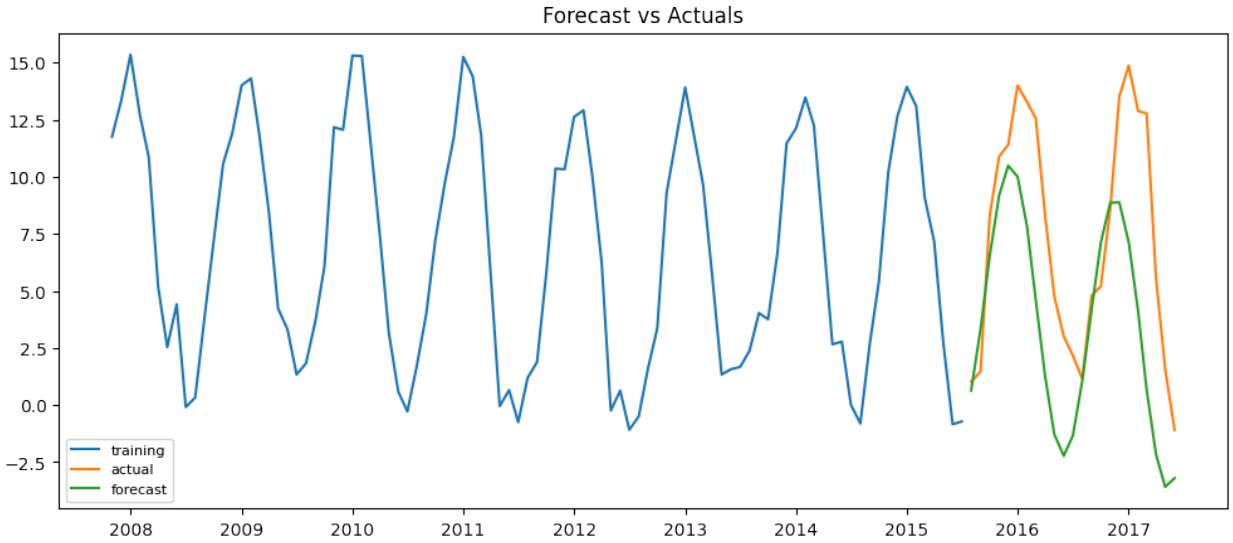
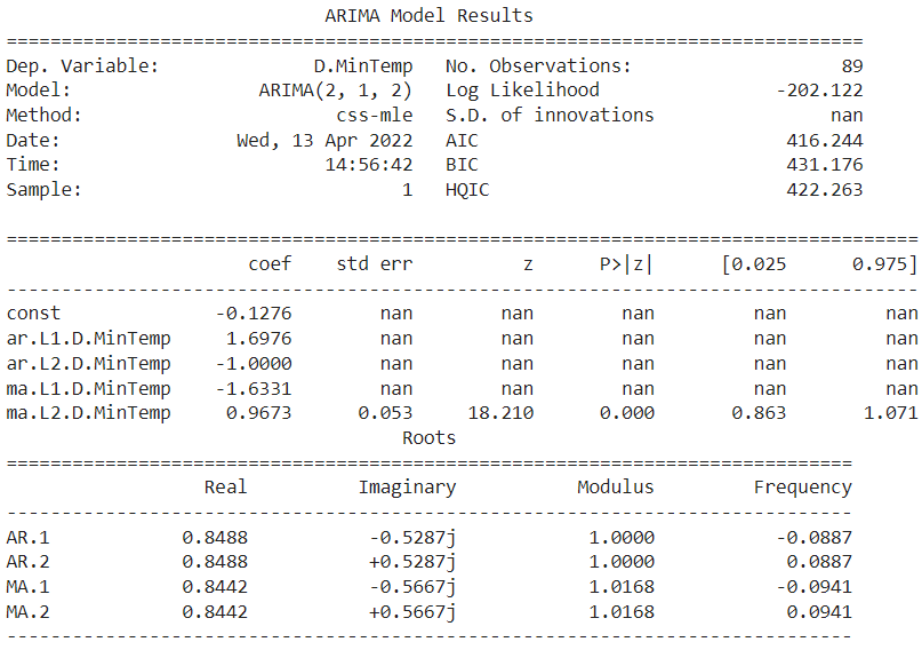


Figura 20: Andamento serie temporale su MinTemp

Per effettuare la cross-val dation, si divide il dataset in train e test, in modo da confrontare il valore predetto con il valore reale. Quindi, per effettuare la divisione, a partire dal dataset di partenza è stato creato un dataset di train con l'80% di elementi in ordine temporale e il restante 20% come dataset di test per il confronto con il valore predetto. Viene impostato un intervallo di confidenza del 95%. Dal grafico in figura 20 si può notare che il modello sembra dare una previsione corretta dal punto di vista della direzione, ma ciascuna delle previsioni è costantemente al di sotto dei valori reali. Per quanto riguarda la descrizione parametrica del modello, in figura 21 si può vedere come non si riescono ad ottenere valori di $P > |z|$. Quindi si può tentare di migliorare il modello aumentando iterativamente i valori di p e q.



ARIMA Model Results

Dep. Variable:	D.MinTemp	No. Observations:	89
Model:	ARIMA(2, 1, 2)	Log Likelihood	-202.122
Method:	css-mle	S.D. of innovations	nan
Date:	Wed, 13 Apr 2022	AIC	416.244
Time:	14:56:42	BIC	431.176
Sample:	1	HQIC	422.263

	coef	std err	z	P>|z|	[0.025	0.975]
const	-0.1276	nan	nan	nan	nan	nan
ar.L1.D.MinTemp	1.6976	nan	nan	nan	nan	nan
ar.L2.D.MinTemp	-1.0000	nan	nan	nan	nan	nan
ma.L1.D.MinTemp	-1.6331	nan	nan	nan	nan	nan
ma.L2.D.MinTemp	0.9673	0.053	18.210	0.000	0.863	1.071

Roots

	Real	Imaginary	Modulus	Frequency
AR.1	0.8488	-0.5287j	1.0000	-0.0887
AR.2	0.8488	+0.5287j	1.0000	0.0887
MA.1	0.8442	-0.5667j	1.0168	-0.0941
MA.2	0.8442	+0.5667j	1.0168	0.0941

Figura 21: Andamento serie temporale su MinTemp

Effettuando diversi tentativi di incrementi dei parametri e osservando sia il modello dal grafico che dai risultati numerici, abbiamo constatato che il miglior cambiamento si ottiene impostando p=3, d=1 e q=3. Il modello che si è ottenuto, presente nella Figura 22, è quindi stato utilizzato per effettuare la previsione sempre sull'intervallo temporale considerato, includendo anche un intervallo di confidenza del 95%. Sebbene considerato che questa previsione risulta essere molto valida (anche considerando i parametri del modello ARIMA che hanno valore basso e l'autocorrelazione dei lag), il modello sembra essere in grado di catturare abbastanza bene l'andamento della serie, il quale rimane effettivamente abbastanza stabile. Anche l'AIC si è ridotto da 416 a 378, il che rappresenta un buon segnale.

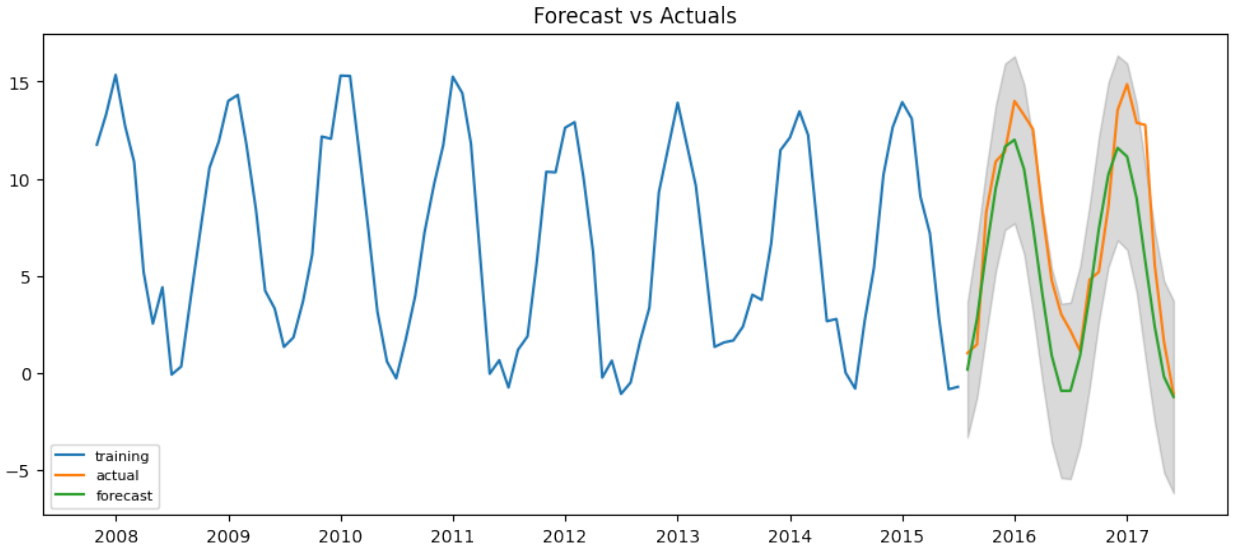


Figura 22: Andamento serie temporale su MinTemp

ARIMA Model Results

Dep. Variable:	D.MinTemp	No. Observations:	89
Model:	ARIMA(3, 1, 3)	Log Likelihood	-181.453
Method:	css-mle	S.D. of innovations	1.779
Date:	Tue, 19 Apr 2022	AIC	378.906
Time:	14:16:37	BIC	398.815
Sample:	1	HQIC	386.930

	coef	std err	z	P>|z|	[0.025	0.975]
const	-0.0227	0.012	-1.826	0.072	-0.047	0.002
ar.L1.D.MinTemp	2.1322	0.125	17.071	0.000	1.887	2.377
ar.L2.D.MinTemp	-1.7195	0.214	-8.045	0.000	-2.138	-1.301
ar.L3.D.MinTemp	0.4220	0.124	3.400	0.001	0.179	0.665
ma.L1.D.MinTemp	-2.5429	0.100	-25.436	0.000	-2.739	-2.347
ma.L2.D.MinTemp	2.3629	0.188	12.577	0.000	1.995	2.731
ma.L3.D.MinTemp	-0.8174	0.099	-8.273	0.000	-1.011	-0.624

Roots

	Real	Imaginary	Modulus	Frequency
AR.1	0.8583	-0.5180j	1.0025	-0.0864
AR.2	0.8583	+0.5180j	1.0025	0.0864
AR.3	2.3579	-0.0000j	2.3579	-0.0000
MA.1	1.0097	-0.0000j	1.0097	-0.0000
MA.2	0.9405	-0.5719j	1.1008	-0.0869
MA.3	0.9405	+0.5719j	1.1008	0.0869

Figura 23: Andamento serie temporale su MinTemp

In tabella 2 sono state riportate le metriche ottenute dal modello. Possiamo notare che il valore Mean Percentage Error e Mean Error sono piuttosto contenuti ottenendo un errore minimo. Il Mean Absolute

Error sembra essere abbastanza valido in quanto il suo valore non è molto elevato. Questo ci dice che non dovremmo aspettarci molta imprecisione dalla previsione. Anche il Mean Absolute Percentage Error assume un valore abbastanza basso e ciò ci permette di affermare che il modello utilizzato potrebbe essere il migliore. Anche il valore Min-Max Error è molto contenuto e questo indica che non ci sono differenze di errori molto grandi ma sono tutti contenuti entro un dato range. Infine, il valore del Root Mean Square Error è relativamente basso. Questa statistica è sempre positiva, con valori più bassi che indicano prestazioni più elevate. L'RMSE può anche essere confrontato con il MAE per vedere se ci sono imprecisioni sostanziali ma non comuni nella previsione. Più ampio è il divario tra RMSE e MAE, più irregolare è la dimensione dell'errore. Nel nostro caso, abbiamo che queste due metriche hanno uno scostamento di un valore pari a 0.5 circa, quindi la previsione risulta essere abbastanza accurata. Per quanto riguarda l'autocorrelazione, i valori sono diversi da zero, ma non perfettamente pari ad 1. Questo, però, ci permette di dire che i dati sono abbastanza correlati tra loro. Mentre per quanto riguarda il CORR, il valore è prossimo ad 1, quindi possiamo dire che si ha una buona correlazione tra i valori di actual e i valori di forecast.

Nome metrica	**Valore**
Lag 1 Autocorrelation of Error (acf1)	0.56114
Correlation between the actual and the forecast (corr)	0.89730
Mean Absolute Error (mae)	2.49197
Mean Absolute Percentage Error (mape)	0.47837
Mean Error (me)	-2.02214
Min-Max Error (minmax)	0.44207
Mean Percentage Error (mpe)	-0.33430
Root Mean Squared Error (rmse)	2.99780

Tabella 2: Metriche ottenute dal modello

Infine, come ultima valutazione, è stato ritenuto opportuno analizzare i residual plots di ARIMA. Nella figura 24 vengono mostrati quattro grafici che descrivono gli errori residui, la densità, l'andamento della distribuzione e il correlogramma. Dal grafico in alto a sinistra si nota che gli errori residui si aggirano intorno ad una media di zero e hanno una varianza uniforme. Nel secondo grafico in alto a destra, il grafico della densità suggerisce una distribuzione normale con lo zero medio. Nel grafico in basso a sinistra, tutti i punti seguono l'andamento della linea rossa, non avendo una deviazione significativa, il quale implicherebbe una distribuzione diversa. Infine, dal correlogramma in basso a destra, è possibile vedere che gli errori residui non sono autocorrelati. Qualsiasi autocorrelazione implicherebbe che ci sia qualche schema negli errori residui il quale non vengono spiegati nel modello.

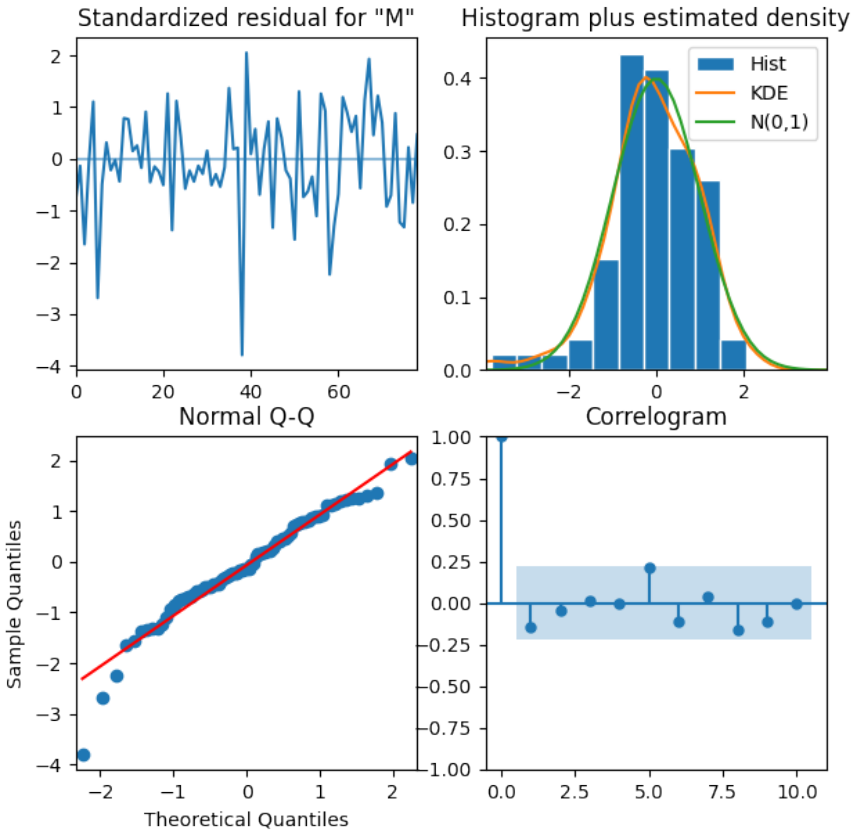

Standardized residual for "M"
Histogram plus estimated density
Hist
KDE
N(0,1)
Normal Q-Q
Correlogram
Sample Quantiles
Theoretical Quantiles

Figura 24:

2 Dataset powerlifting

Per effettuare la regressione e classificazione, è stato necessario identificare un dataset idoneo con dati che possono essere classificati e predetti, sia con variabili continue (per la regressione) che con variabili discrete (per la classificazione). La nostra scelta è ricaduta sul dataset "Powerlifting-database" disponibile su Kaggle al seguente link kaggle.com/datasets/dansbecker/powerlifting-database), un dataset che contiene un archivio di pubblico dominio sul powerlifting in tutto il mondo. Il powerlifting è uno sport in cui i concorrenti competono per sollevare il peso maggiore per la loro classe di appartenenza in tre tipi di sollevamenti con bilanciere: lo squat, la panca e lo stacco da terra.

2.1 Struttura del dataset

Il dataset utilizzato si compone di due tabelle: meets e openpowerlifting. La prima contiene informazioni inerenti alle gare di powerlifting mentre la seconda contiene i dati di tutti gli atleti che vi hanno partecipato. Successivamente viene eseguito il merge dei due file in modo tale da avere un unico dataset con tutte le informazioni. Il dataset è composto da diversi campi, essi vengono mostrati nella seguente tabella:

MeetID	Codice che identifica l'atleta.
Name	Il nome e cognome dell'atleta.
Sex	Il sesso dell'atleta.
Equipement	Il tipo di attrezzatura indossata. Le federazioni possono dividere la competizione per attrezzature diverse.
Age	Età dell'atleta.
Division	Divisione dove l'atleta partecipa.
BodyweightKg	Il peso corporeo dell'atleta in Kg.
WeightClassKg	La classe di peso in cui ha gareggiato l'atleta. Le federazioni possono suddividere la competizione in diverse classi di peso.
Squat4Kg	Il quarto tentativo di squat dell'atleta in Kg. Negativo significa che il tentativo è fallito. Ignora NaN. Non comunemente usato.
BestSquatKg	Il miglior squat tra i vari tentavi[Squat1Kg, Squat2Kg, Squat3Kg]. Questo valore verrà conteggiato per il campo TotalKg.
Bench4Kg	Il quarto tentativo di panca dell'atleta in Kg. Negativo significa che il tentativo è fallito. Ignora NaN. Non comunemente usato
BestBenchKg	La migliore alzata di bench tra i vari tentativi [Bench1Kg, Bench2Kg, Bench3Kg]. Questo valore verrà conteggiato per il campo TotalKg.
Deadlift4Kg	Il quarto tentativo di stacco da terra dell'atleta in Kg. Negativo significa che il tentativo è fallito. Ignora NaN. Non comunemente usato
BestDeadliftKg	Il miglior stacco da terra tra i vari tentativi [Deadlift1Kg, Deadlift2Kg, Deadlift3Kg]. Questo valore verrà conteggiato per il campo TotalKg.
TotalKg	La somma di BestSquatKg, BestBenchKg, BestDeadliftKg.
Place	Posizionamento nella gara.
Wilks	Coefficiente usato come formula per eleggere il Best Lifter nella sua categoria.
MeetPath	Codice della gara.
Federation	Federazione che ospita la gara.
Date	Data in cui l'atleta gareggia.
MeetCountry	Nazione che ospita la gara.
MeetState	Regione che ospita la gara.
MeetTown	Città che ospita la gara.
MeetName	Nome della gara.

2.2 Pulizia del dataset

Successivamente viene analizzato il dataset per identificare quali campi contengono valori "Nan" o non contengono valori. In particolare, viene implementata una semplice funzione che mostra in output la percentuale di valori "Nan" su ogni campo del dataset. Tanto più è elevata la percentuale, tanto maggiore sarà la quantità di dati mancanti sui rispettivi campi. L'immagine 25 mostra quali campi del dataset possiedono maggiormente campi vuoti.

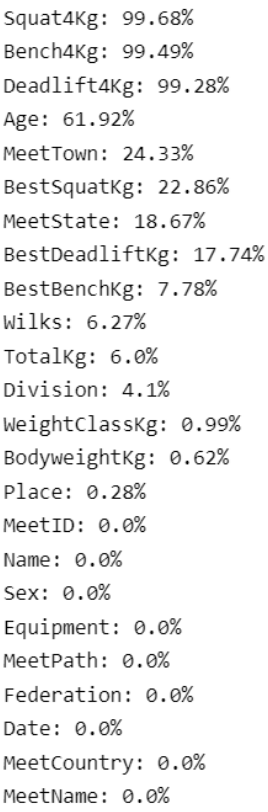


Figura 25: Percentuale dei dati mancanti su ogni campo

Al fine di completare i task oggetto di questo progetto, vengono implementate le seguenti operazioni di pulizia del dataset:

- il campo "Age" è stato troncato;
- nei campi "BestSquatKg", "BestBenchKg" e "BestDeadliftKg" sono stati considerati solo i valori maggiori di zero, in quanto alcuni di esse presentavano valori negativi significando il fallimento dello squat, della panca o degli stacchi;
- i campi "Squat4Kg", "Deadlift4Kg" e "Bench4Kg" sono stati eliminati perché inutili e vuoti oltre il 99% dei casi.

Attraverso il comando `openpowerlifting.describe()` si ottiene in output la tabella in figura 26 in cui si visualizzano delle statistiche descrittive che riassumono la tendenza centrale, la dispersione e la forma della distribuzione di un set di dati, escludendo i valori NaN.

	MeetID	Sex	Age	BodyweightKg	BestSquatKg	BestBenchKg	BestDeadliftKg	TotalKg	Wilks
count	286645.000000	286645.000000	106827.000000	286046.000000	286645.000000	286645.000000	286645.000000	286191.000000	285598.000000
mean	5187.833787	0.242896	29.673388	85.511386	177.558843	114.643489	195.511315	487.758391	347.808746
std	2508.242237	0.428834	11.772323	22.576177	65.621992	47.883186	58.290919	165.089864	77.317373
min	0.000000	0.000000	7.000000	23.900000	13.600000	6.800000	2.270000	38.600000	28.230000
25%	3340.000000	0.000000	21.000000	68.500000	127.500000	75.000000	149.690000	356.070000	295.560000
50%	6016.000000	0.000000	26.000000	82.200000	174.630000	112.500000	195.000000	485.000000	344.245000
75%	7073.000000	0.000000	35.000000	99.200000	217.720000	145.000000	237.500000	597.500000	394.577750
max	8481.000000	1.000000	93.000000	242.400000	573.790000	455.860000	440.000000	1365.310000	779.380000

Figura 26: Output del comando `openpowerlifting.describe()`

2.3 Visualizzazione dei dati

2.3.1 Distribuzioni uomini e donne

Con il seguente grafico si vuole analizzare la distribuzione del sesso fra i vari record presenti nel dataset. Come si può vedere dal grafico a torta, 3/4 dei dati riguarda gli uomini mentre solo 1/4 è sulle donne.

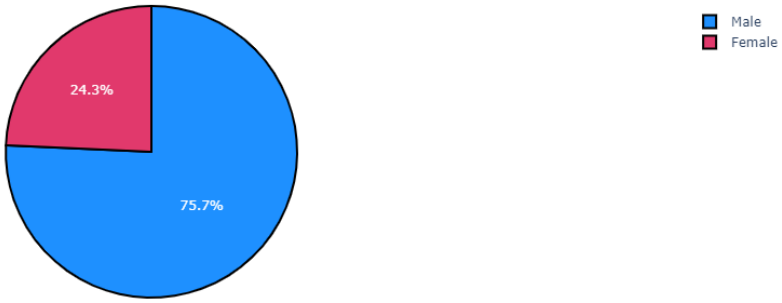


Figura 27: Distribuzione uomini e donne sull'intero dataset

2.3.2 Distribuzioni dell'età degli atleti

In questo grafico si visualizza la distribuzione dell'età fra i vari atleti. Si nota che la maggior parte di questi sono molto giovani, con un età compresa tra i 18 e 30 anni. Interessante notare che ci sono anche atleti con meno di 14 anni, insieme ad atleti con più di 70 anni.

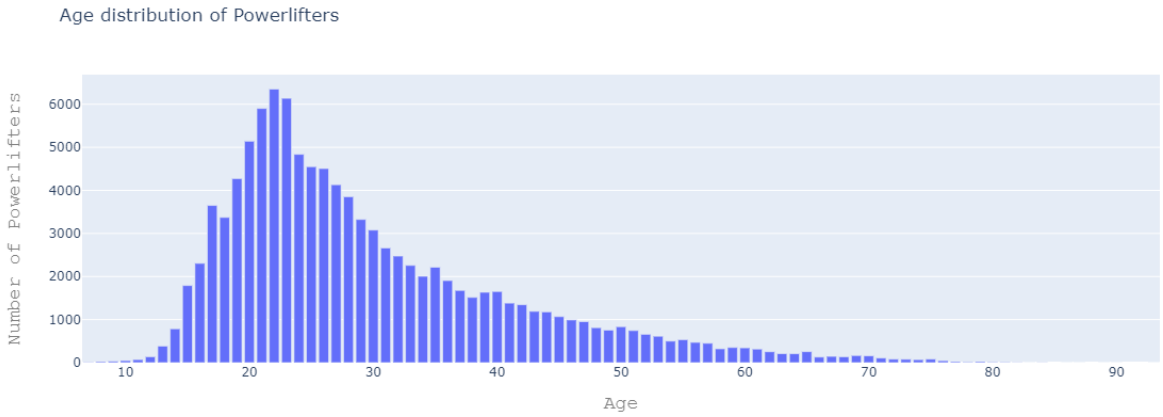


Figura 28: Distribuzione dell'età degli atleti

2.3.3 Distribuzioni del peso degli atleti

In questo paragrafo è stato ritenuto interessante mostrare tramite un conteggio il numero di atleti che hanno uno specifico peso. Come si può notare dalla figura 29 la maggior parte degli atleti ha un peso che varia tra i 50 Kg e 120 kg.

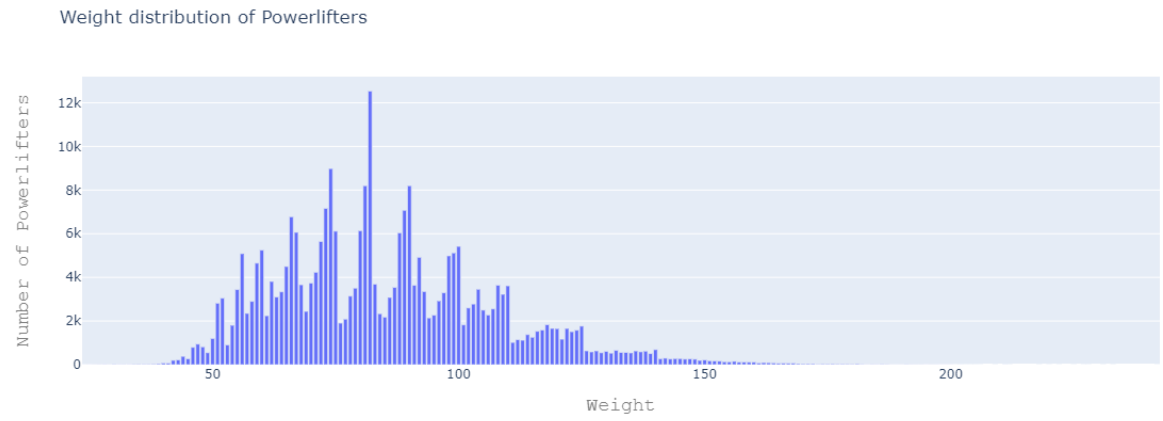


Figura 29: Distribuzione del peso degli atleti

2.3.4 Età degli atleti su ogni competizione

In questo sezione viene analizzata l'età degli atleti rispetto a determinati parametri, che sono: per il Deadlift viene considerata l'età di ogni atleta e il peso migliore che essi hanno sollevato, quindi ci si può fare un'idea di qual è la fascia di età in cui viene sollevato il maggior peso; lo stesso viene fatto sia per lo Squat che per la BenchPress. Invece, nel grafico in basso a destra viene rappresentato nelle ascisse l'età dell'atleta e nelle ordinate si ha Total Kgs che è dat dalla somma dei migliori valori di Deadlift, Squat e Benchpress. Come si può vedere dai grafici gli atleti più forti si hanno nella fascia di età compresa tra 20 e 40 anni.

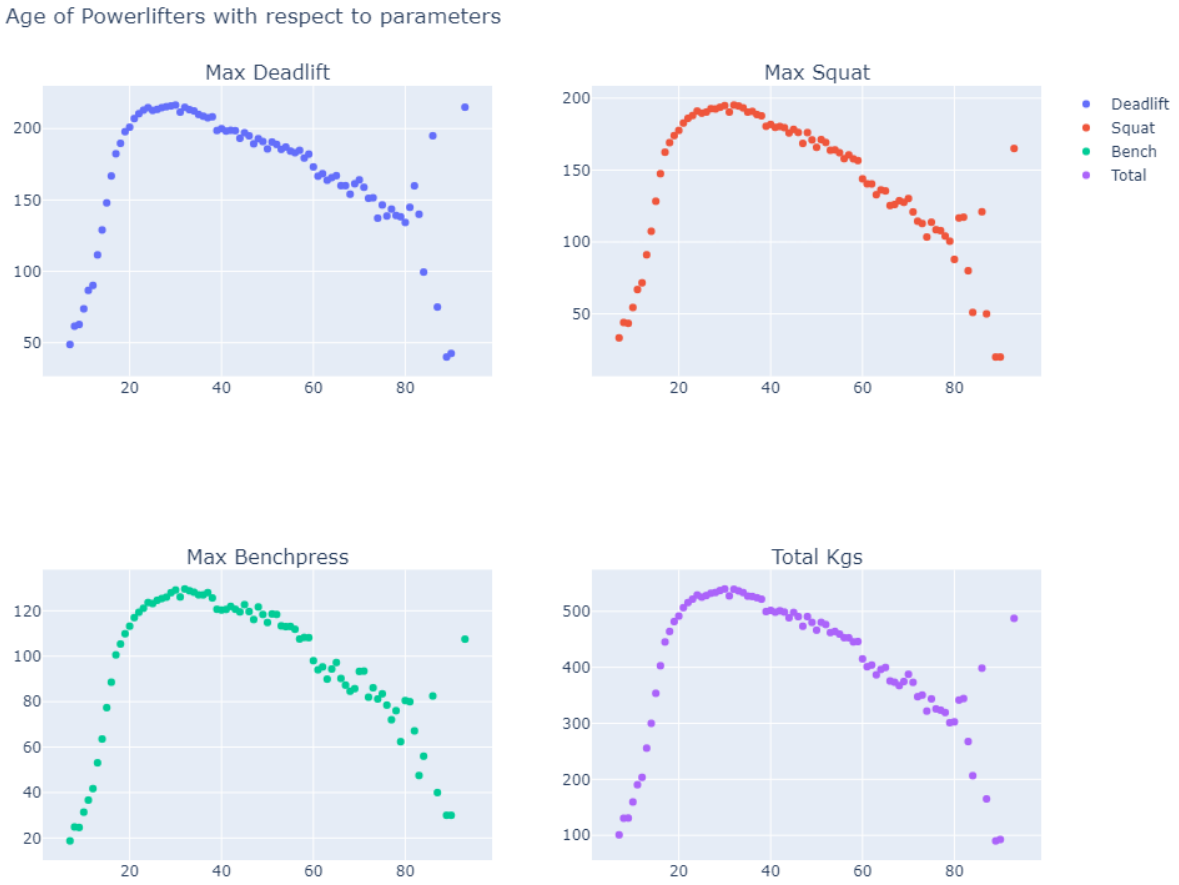


Figura 30: Distribuzione età atleti su ogni tipologia di gara

2.3.5 Coefficiente wilks in base all'età degli atleti

Nel grafico in figura 31 si può visualizzare l'andamento del coefficiente wilks in base all'età. Si nota che a partire dalla giovanissima età di 9-10 anni, questo sale molto rapidamente per arrestare la sua crescita circa a 30 anni. Da questa età, il coefficiente inizia lentamente a decrescere fino agli 80 anni, dove da qui in poi inizia a oscillare in un range molto ampio.

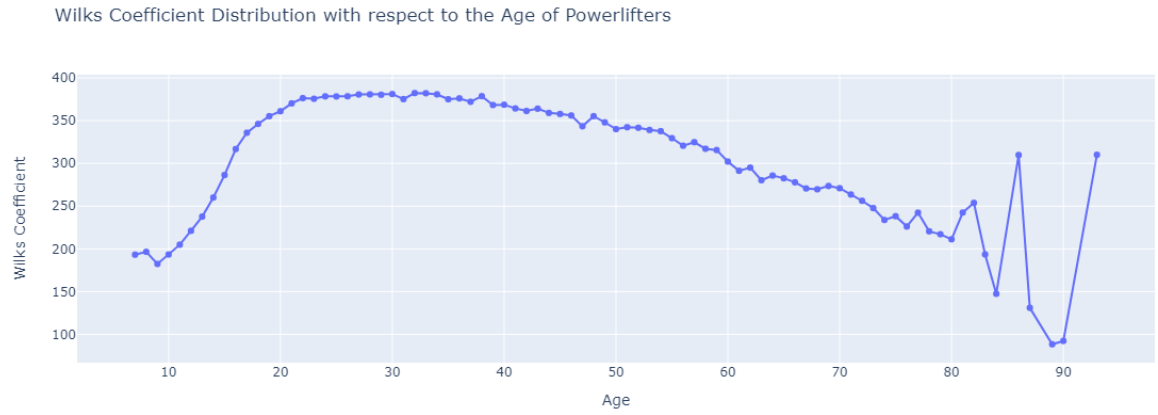


Figura 31: Distribuzione del coefficiente wilks sull'età

2.3.6 Coefficiente wilks in base al peso degli atleti

In questa sezione viene analizzato il coefficiente Wilks rispetto al peso e si è scelto di tenere come valore di mezzo 300. Come si può notare dal grafico si ha che la maggioranza degli atleti ha un punteggio superiore a 300, mentre gli atleti con punteggio inferiore a 300 sono in quantità minore.

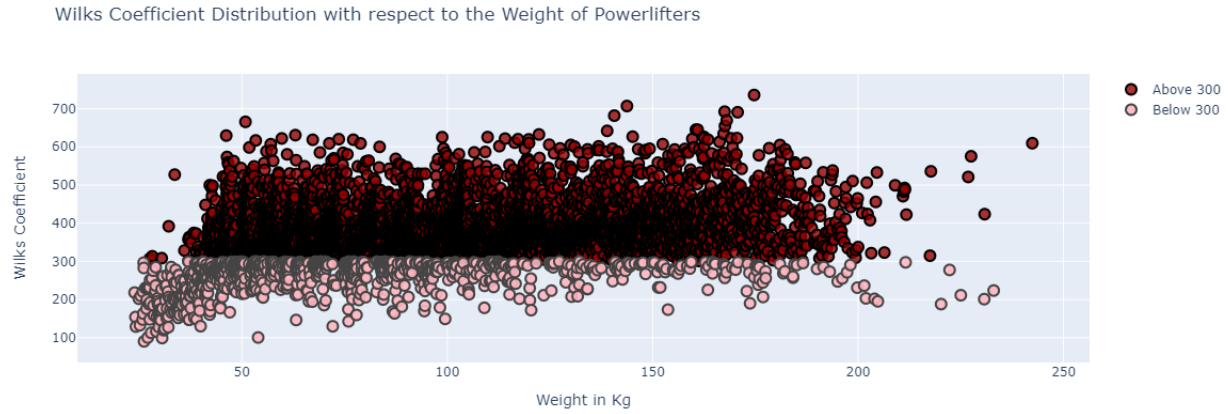


Figura 32: Distribuzione del coefficiente wilks sul peso

2.3.7 Squat migliore

In questo paragrafo si va ad esaminare lo squat migliore fatto dall'atleta rispetto all'età. Il risultato ottenuto dall'analisi è interessante ed è congruo coi risultati ottenuti dai grafici sopra. Infatti, come si può osservare, gli atleti con età tra 20 e 40 sono gli atleti che hanno eseguito gli squat migliori e di conseguenza hanno sollevato il peso maggiore.

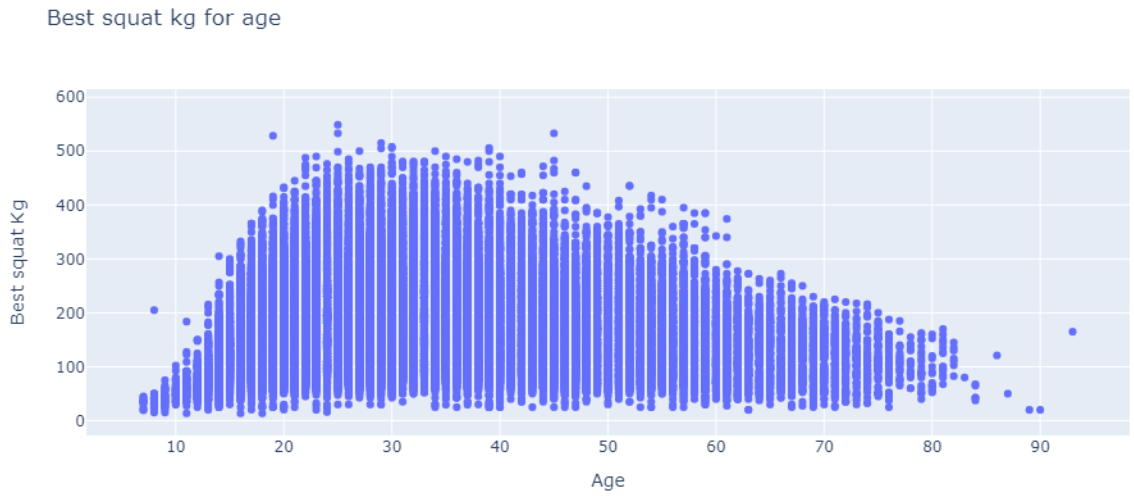


Figura 33: Squat migliore per età

2.3.8 Matrice di correlazione

Infine, è stata analizzata la correlazione tra 6 campi, i quali sono: Age, BodyweightKg, BestSquatKg, BestBenchKg, BestDeadliftKg e Wilks. Essi vengono studiati attraverso una matrice di correlazione, più il valore è in prossimità di 1 o -1 e più i dati sono correlati positivamente o negativamente. Si può notare dalla matrice, attraverso il valore numerico, che molti campi hanno una buona correlazione tra loro. I restanti campi precedentemente descritti sono stati eliminati in quanto contengono valori non numerici o non vengono ritenuti utili ai fini dei task di clustering e regressione.

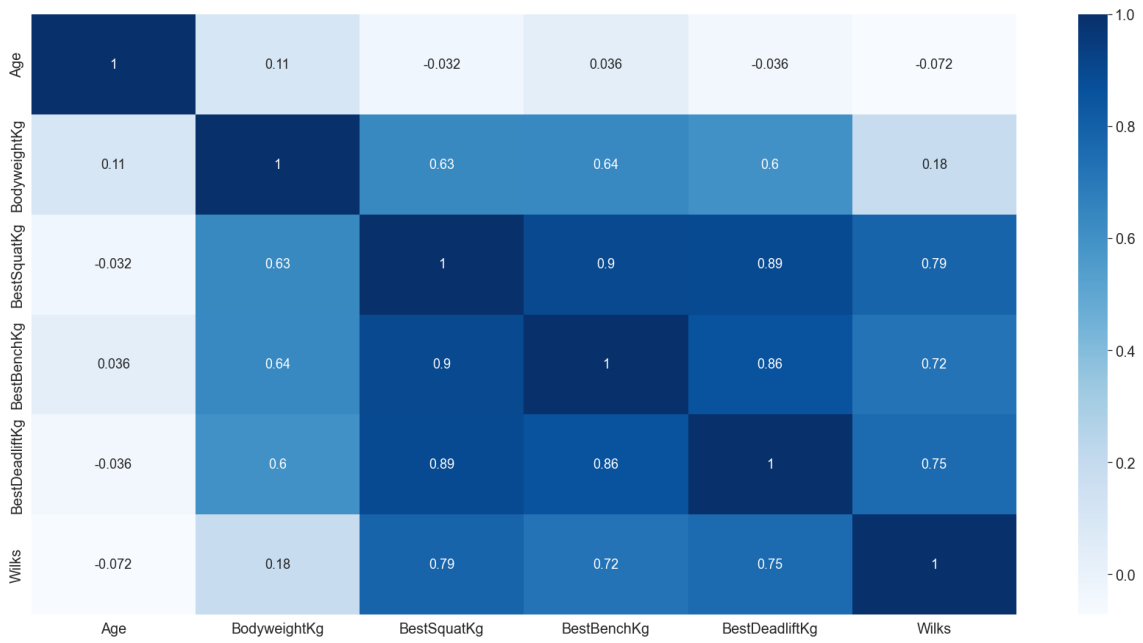


Figura 34: Matrice di correlazione

2.4 Cluster

2.4.1 K-Means

Uno degli algoritmi utilizzati per il clustering è il k-means. L'obiettivo dell'algoritmo è minimizzare la varianza totale intra-gruppo; ogni gruppo viene identificato mediante un centroide o punto medio. L'algoritmo determina casualmente k punti di riferimento, e poi segue iterativamente le seguenti fasi:

1. associa ogni elemento del dataset (sample) al punto di riferimento più vicino formando così k cluster;
2. per ogni cluster ottenuto calcola il centroide, che diventa il nuovo punto di riferimento;

La procedura prosegue finché la posizione dei centroidi non converge. Per trovare il numero k ottimale si è fatto uso del metodo del gomito (Elbow method), in cui si analizza il valore di inerzia relativo al modello in funzione di k. Il miglior k si colloca, infatti, in corrispondenza del punto del grafico in cui è visibile una variazione di pendenza repentina.

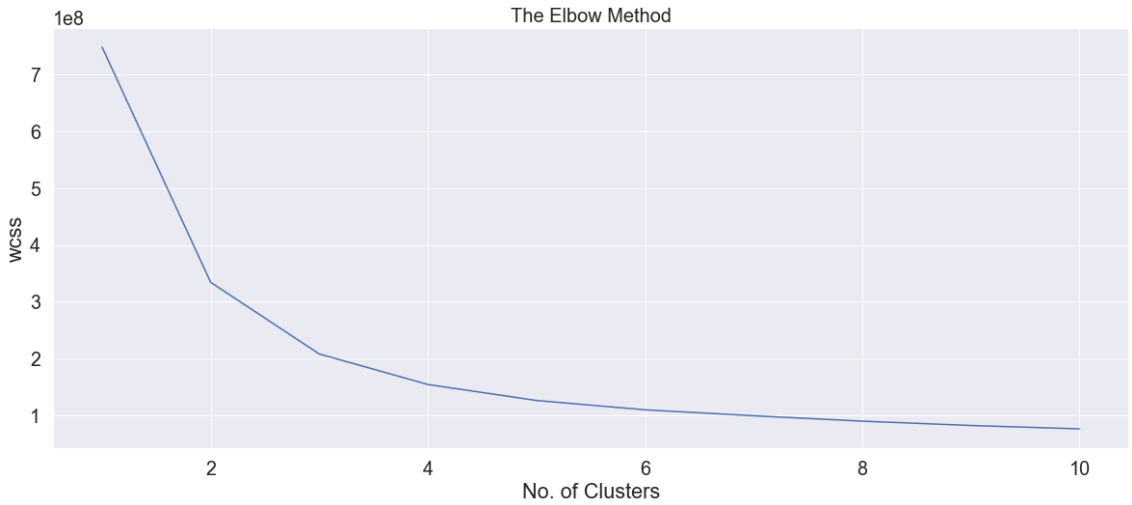


Figura 35: Elbow method

La figura 35 mostra il grafico relativo all'Elbow method. In questo caso si può notare una particolare regolarità del grafico, dalla quale è difficile determinare il valore k. Nonostante il profilo della funzione fa intendere la scarsa propensione del dataset al clustering si è comunque scelto un valore di k pari a 4. A conferma del fatto che il numero dei cluster fosse 4, è stata utilizzata la libreria yellowbrick il quale implementa un metodo che permette di ricavare automaticamente il numero esatto di cluster, come visibile nella figura 36.

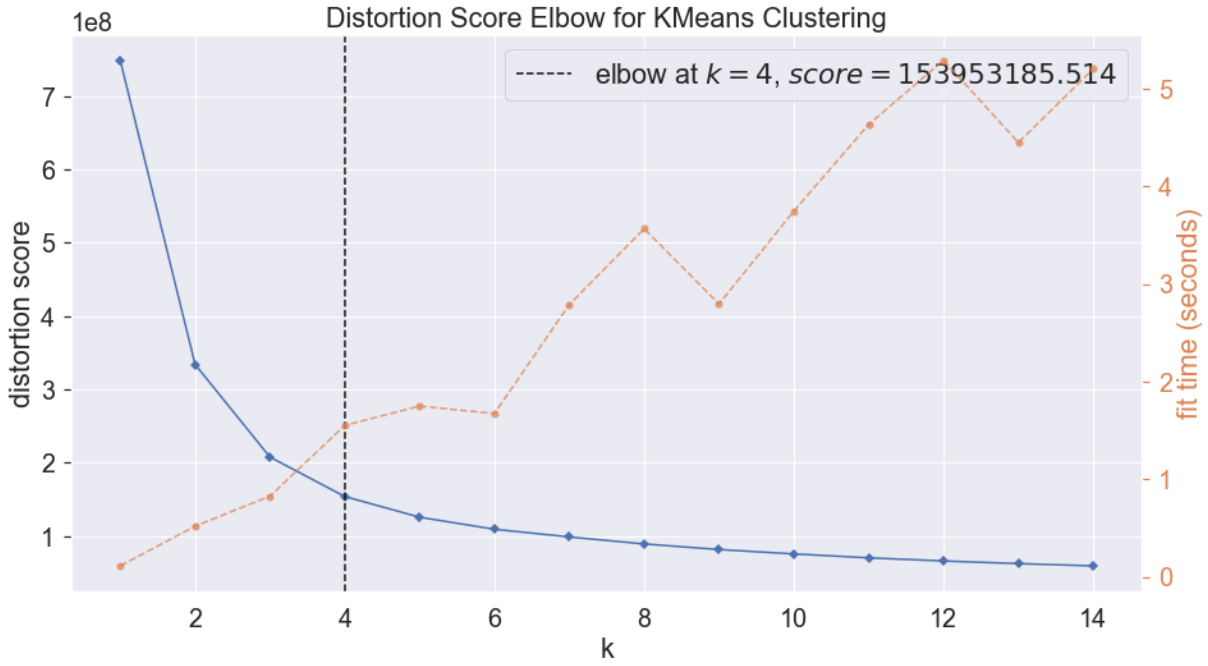


Figura 36: Yellowbrick Elbow method

Nella figura 37 è riportata la clusterizzazione ottenuta dal k-means nella spazio definito dalle dimen-

sioni: Bodyweight, Age e Wilks. I centroidi trovati dall'algoritmo sono evidenziati in rosso all'interno dei grafici. Pur essendo queste le migliori clusterizzazioni possibili da applicare al dataset, non forniscono alcun contributo informativo rilevante.

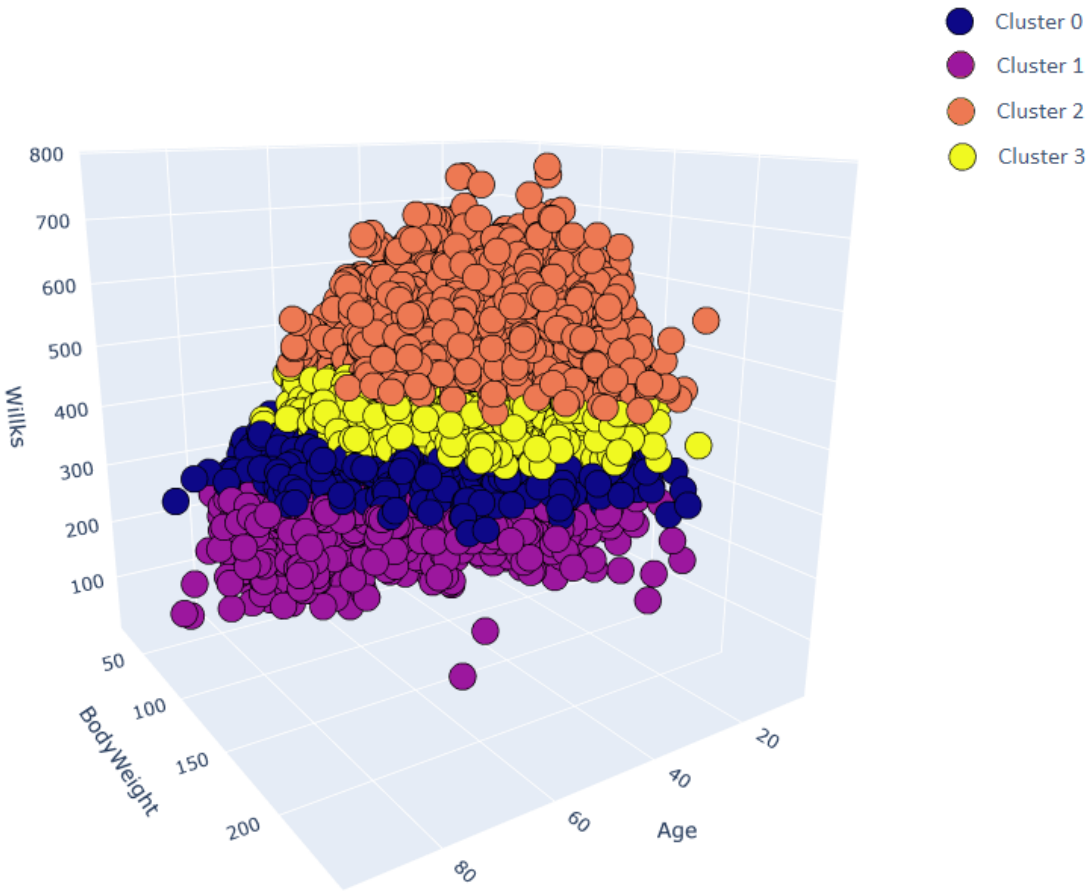


Figura 37: Risultato del K-Means

Analizzando i vari cluster ottenuti, possiamo osservare i diversi range di dati per ogni cluster.

In relazione ai dati del cluster 0, dalla descrizione ottenuta in figura 38, è possibile osservare che l'età va da un minimo di 7 anni a un massimo di 93 (quindi comprende quasi tutto il range), Bodyweight va da 28.20 a 217.5 kg (anche qui un ampio range) e infine Wilks va da 290.98 a 375.86. Nel cluster sono contenuti 40745 record ed è il primo cluster più grande.

	Age	BodyweightKg	Wilks
count	40746.000000	40746.000000	40746.000000
mean	29.106587	82.912051	336.207282
std	11.483234	20.978110	21.778936
min	7.000000	28.200000	290.980000
25%	21.000000	67.000000	317.970000
50%	26.000000	80.970000	337.280000
75%	35.000000	96.000000	354.856000
max	93.000000	217.500000	375.860000

Figura 38: Informazioni sui dati del cluster 0

In relazione ai dati del cluster 1, dalla descrizione ottenuta in figura 39 è possibile osservare che l'età va da un minimo di 7 anni a un massimo di 90 (quindi comprende quasi tutto il range), Bodyweight va da 24.10 a 217.5 kg (anche qui un ampio range) e infine Wilks va da 28.23 a 298.48. Nel cluster sono contenuti 20940 record ed è il terzo cluster in ordine di grandezza.

	Age	BodyweightKg	Wilks
count	20940.000000	20940.000000	20940.000000
mean	32.220248	78.337913	254.664962
std	16.159658	20.064590	34.126295
min	7.000000	24.100000	28.230000
25%	19.000000	64.417500	236.442250
50%	27.000000	74.800000	263.055000
75%	42.000000	89.300000	281.472500
max	90.000000	220.200000	298.480000

Figura 39: Informazioni sui dati del cluster 1

In relazione ai dati del cluster 2, dalla descrizione ottenuta in figura 40 è possibile osservare che l'età va da un minimo di 14 anni a un massimo di 65 (un range già più ristretto rispetto ai cluster precedenti), Bodyweight va da 33.66 a 242.4 kg (un ampio range di valori) e infine Wilks va da 455.86 a 779.38. Nel cluster sono contenuti 11314 record ed è il cluster più piccolo.

	Age	BodyweightKg	Wilks
count	11314.000000	11314.000000	11314.000000
mean	30.040658	94.594406	510.594870
std	8.361429	28.509869	45.087188
min	14.000000	33.660000	455.860000
25%	24.000000	73.300000	475.422500
50%	28.000000	91.700000	497.500000
75%	35.000000	110.700000	534.045000
max	65.000000	242.400000	779.380000

Figura 40: Informazioni sui dati del cluster 2

In relazione ai dati del cluster 3, dalla descrizione ottenuta in figura 41 è possibile osservare che l'età va da un minimo di 13 anni a un massimo di 72 (un range piuttosto ampio), Bodyweight va da 40.73 a 230.8 kg (un ampio range di valori) e infine Wilks va da 363.22 a 461.65. Nel cluster sono contenuti 33705 record ed è il secondo cluster più grande.

	Age	BodyweightKg	Wilks
count	33705.000000	33705.000000	33705.000000
mean	28.656134	89.321090	408.784681
std	9.397707	22.781691	24.086453
min	13.000000	40.730000	363.220000
25%	22.000000	73.380000	388.160000
50%	26.000000	88.350000	405.720000
75%	33.000000	103.100000	427.720000
max	72.000000	230.800000	461.650000

Figura 41: Informazioni sui dati del cluster 3

Nell'immagine 42, il valore medio del coefficiente di silhouette è pari a circa 0.38 e molti elementi di ogni cluster supera tale valore o comunque ha una silhouette positiva. Questo ci consente di dire che gli elementi dei vari gruppi sicuramente sono ben rappresentati dai cluster.

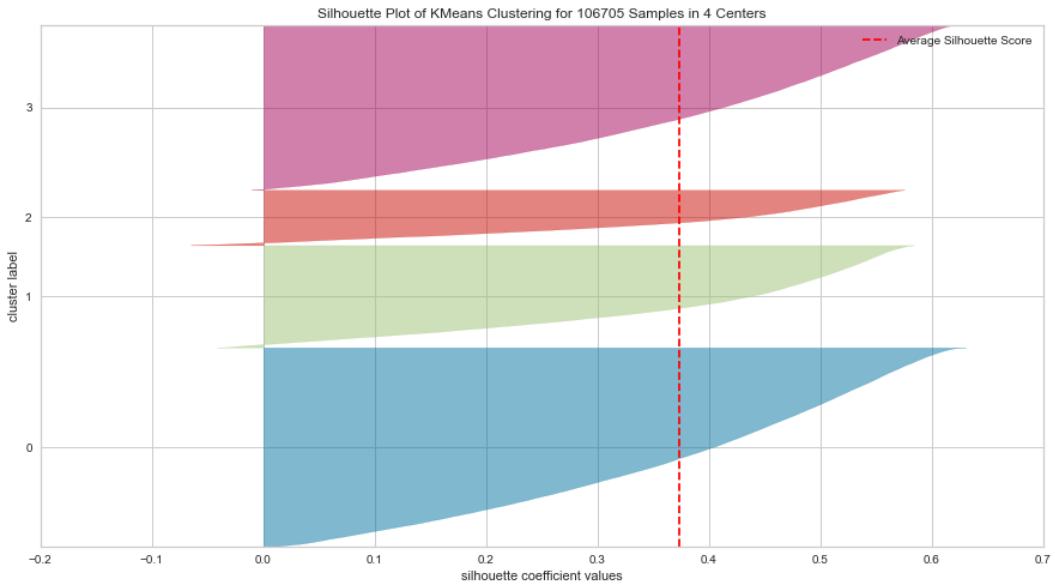


Figura 42: Grafico sulla silhouette

2.4.2 Cluster con PCA

Successivamente, viene eseguita l'operazione di clustering con PCA sfruttando anche altri attributi a nostra disposizione. In questo caso, passando da una situazione tridimensionale ad n-dimensionale, per riuscire a visualizzare i grafici si è scelto di utilizzare il metodo della PCA (Principal Component Analysis). Questo metodo ci permette di ridurre il numero di variabili misurate mantenendo il più possibile intatta la conoscenza iniziale. Nuovamente, l'operazione di clustering è stata effettuata con l'algoritmo "K-Means" e con un numero di cluster pari a 4.

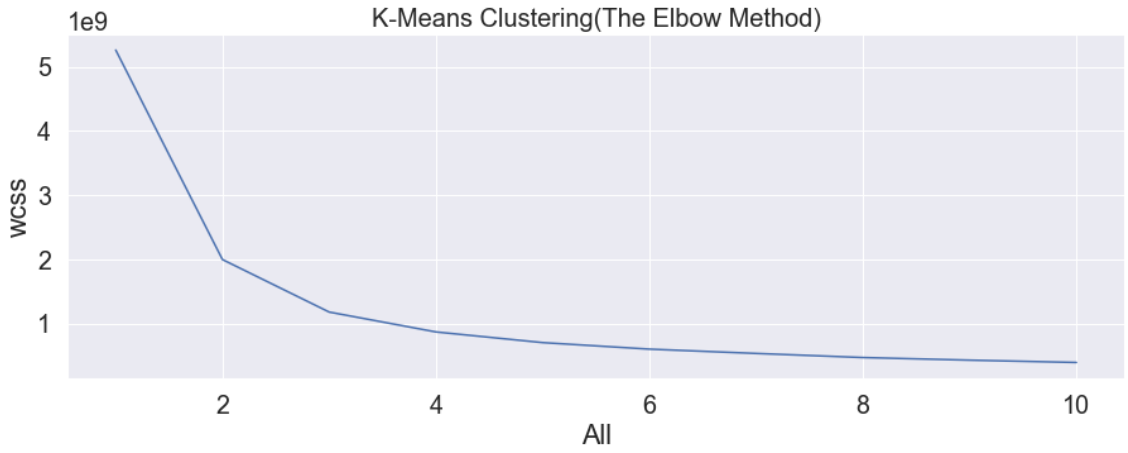


Figura 43: Elbow method

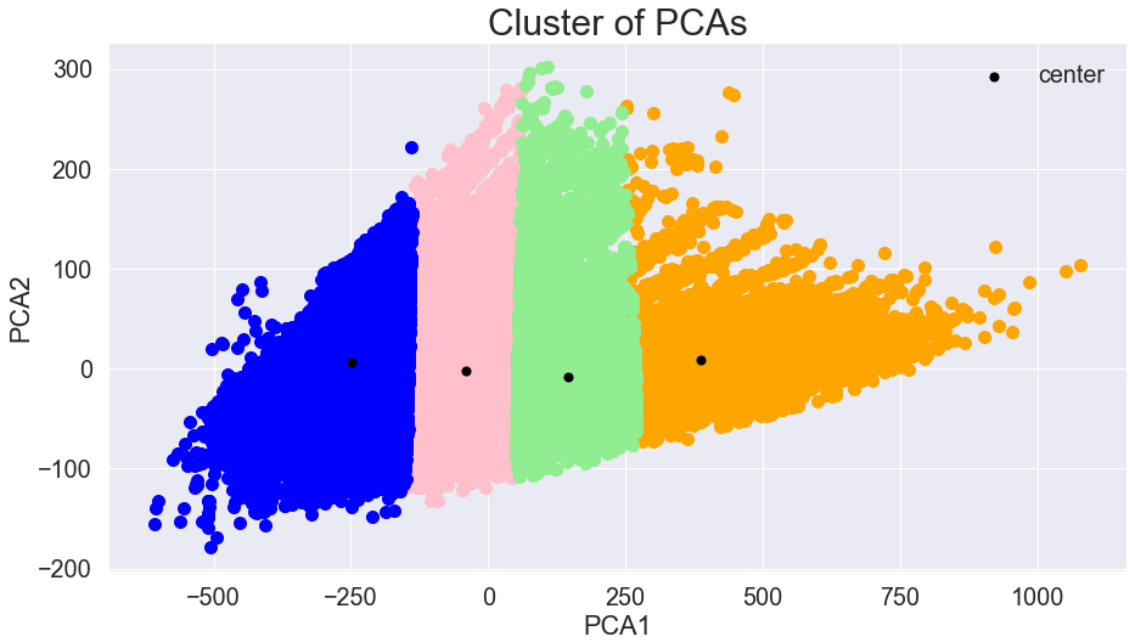


Figura 44: Risultato del clustering con PCA

In questo caso, la silhouette media risulta essere pari a circa 0.41 e viene superata da abbastanza elementi di ogni cluster. Gli elementi dei vari gruppi vengono rappresentati discretamente dai cluster, infatti, come si evince dal grafico, la maggior parte di essi ha un valore positivo di silhouette.

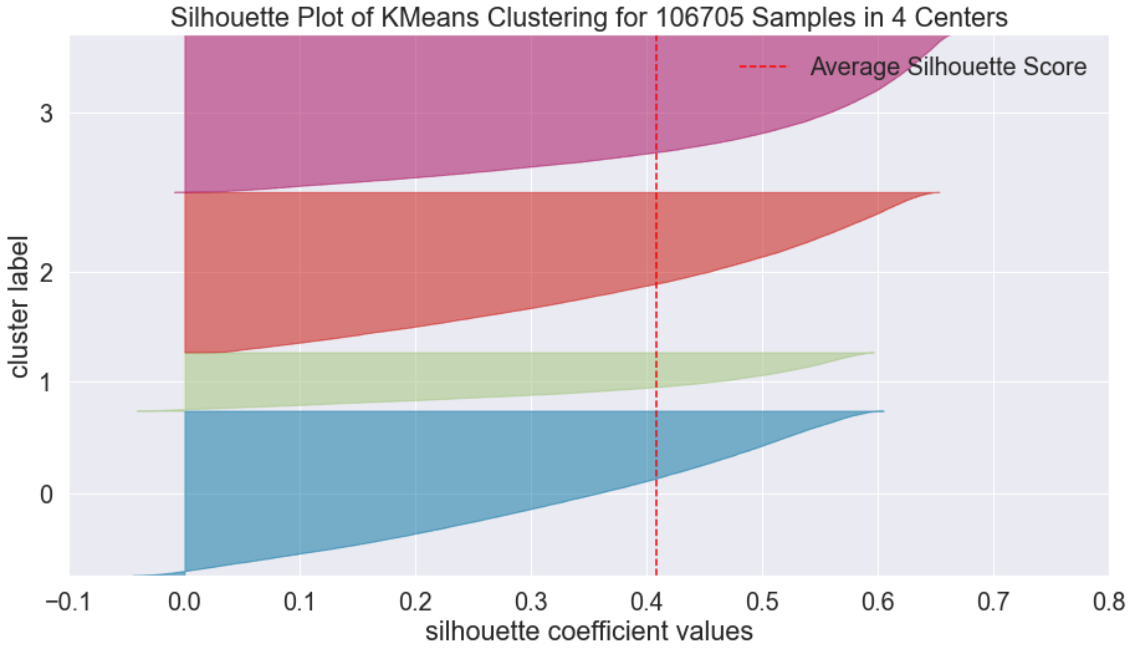


Figura 45: Grafico sulla silhouette con PCA

2.4.3 DB-SCAN

Come ultimo test, viene effettuata un'operazione di clustering con PCA utilizzando l'algoritmo DB-SCAN. Il DB-SCAN è un altro algoritmo per la clusterizzazione basato sul concetto di densità. Esso necessita di due parametri: ϵ e η. L'algoritmo distingue 3 tipi di punti:

- punti interni: che hanno almeno n punti a distanza inferiore di ϵ;
- punti di confine: che non sono interni, ma hanno almeno un punto interno a distanza inferiore di ϵ;
- punti di rumore: che non sono ne interni, ne di confine.

L'algoritmo, a differenza del K-Means, si conclude dopo una sola scansione. Durante il processo ogni punto viene riconosciuto in una delle 3 tipologie. Tutti i punti interni adiacenti (distanti meno di ϵ tra loro) vengono considerati parte dello stesso cluster. Ogni punto di confine viene associato al cluster che contiene il suo punto interno adiacente, mentre il rumore viene ignorato. Anche in questo caso l'incompatibilità del dataset al task di clusterizzazione è evidente già a partire dalla scelta dei parametri ϵ e η. Per trovare il valore di ϵ ottimale rispetto ad un η scelto, si definisce un grafico in cui sulle ascisse vengono rappresentati tutti i punti del dataset e sulle ordinate la distanza che separa ciascuno dall'ennesimo elemento più vicino.

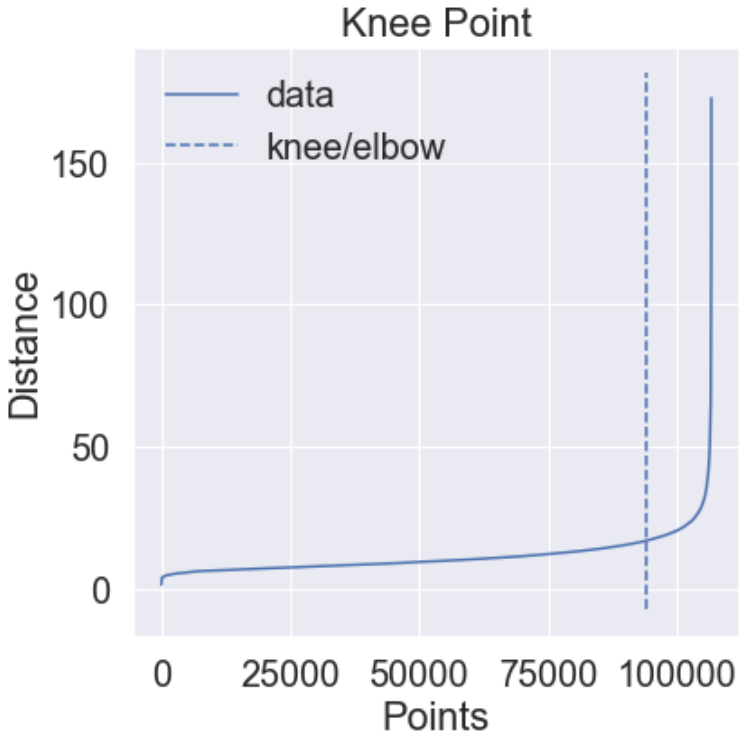


Figura 46: Grafico Nearest Neighbors

Il valore di ϵ, analogamente a quanto visto con l'Elbow method, viene determinato dal punto in cui si presenta una netta variazione di pendenza: la y relativa a quel punto è la ϵ ottimale.

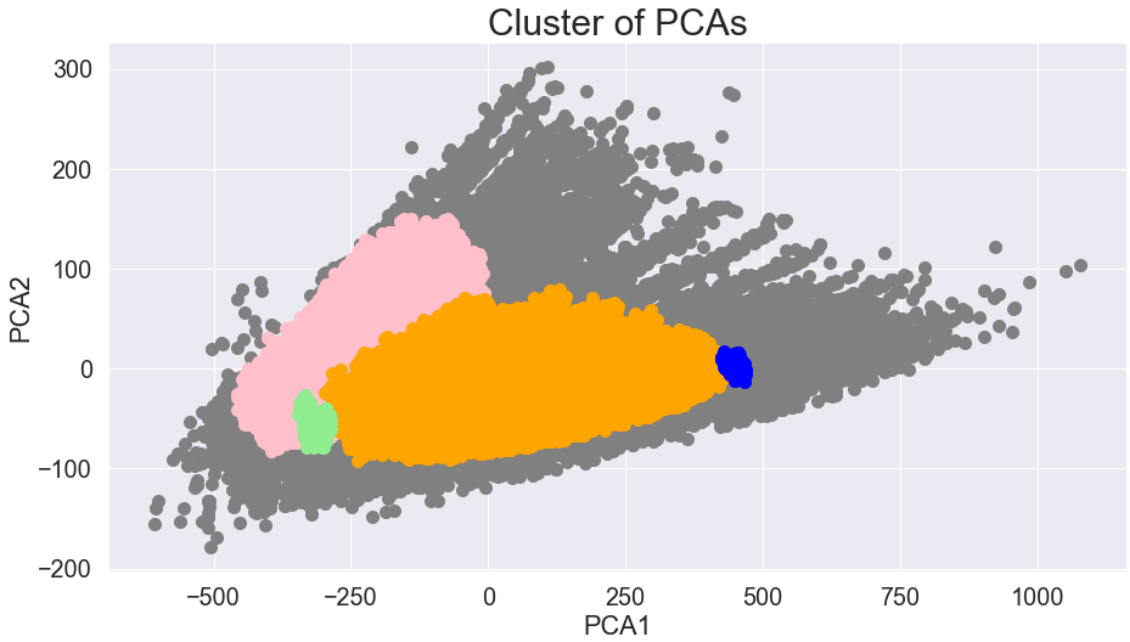


Figura 47: Cluster con DBSCAN sfruttando la PCA

Per effettuare l'operazione di clustering con algoritmo DB-SCAN definiamo un epsilon pari a 17. Come è possibile notare dalla figura 47, i cluster visualizzati sono ben distinti, anche se alcuni punti si sovrappongono, non essendo molto chiari. Inoltre, ci sono tantissimi noise-point che corrispondono a punti colorati in grigio scuro.

2.5 Regressione

Per lo sviluppo del task di regressione si è fatto uso della libreria sklearn. Abbiamo deciso di effettuare una previsione su BestsquatKg, ovvero il migliore tentativo di squat di un atleta, a partire dal sesso, peso, età e il coefficiente wilks. Per il task sono stati allenati diversi regressori, i cui risultati sono stati poi confrontati usando come metriche principali l'R-squared, il MAE, l'RMSE e l'MSE. I regressori considerati sono i seguenti:

1. Un regressore lineare;
2. Un regressore di tipo random forest;
3. Un regressore che fa uso dell'SVM, con kernel polinomiale;
4. Un regressore di tipo gradient boosting;
5. Un regressore di tipo adaboost.

Prima di iniziare l'allenamento dei vari modelli i dati in ingresso sono stati processati, effettuando uno split del dataset tra training set e test set con una divisione 70% / 30%, e effettuando in seguito una normalizzazione dei valori. Tramite l'utilizzo della matrice di correlazione si possono vedere alcuni campi troppo correlati e quindi abbiamo ritenuto opportuno eliminarli in quanto non ottimali per predire correttamente BestsquatKg.

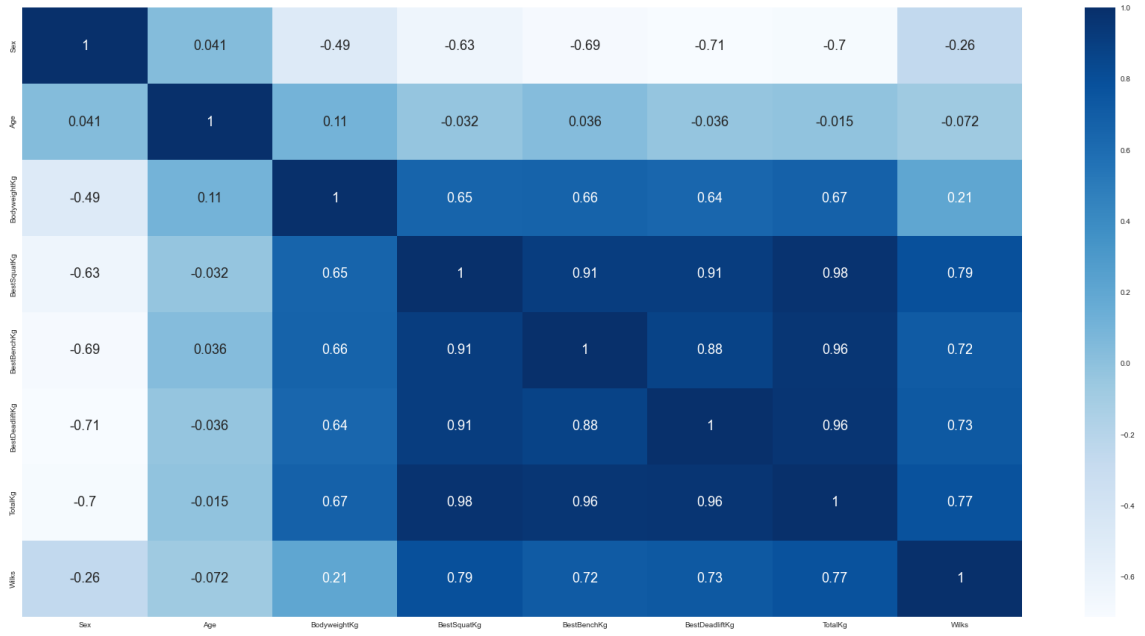


Figura 48: Matrice di correlazione

Nella figura 49 viene mostrata la matrice di correlazione con i campi eliminati. Possiamo vedere come i livelli di correlazione sono abbastanza lievi, ma comunque utili per fare regressione.

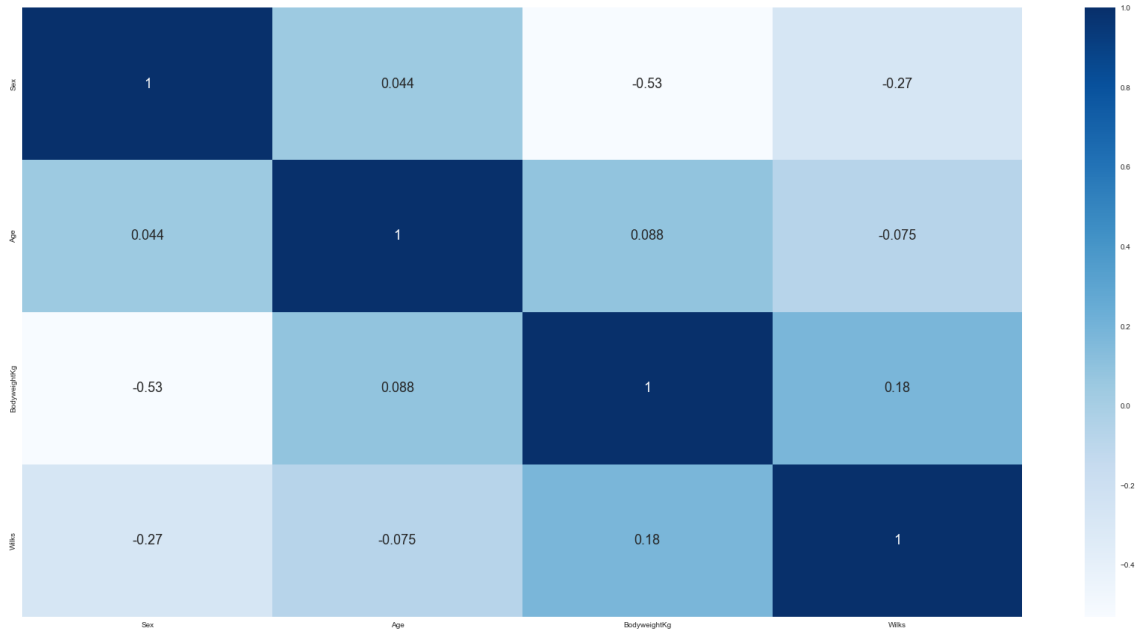


Figura 49: Matrice di correlazione con campi eliminati

Dopo questa fase di preprocessing si è potuto quindi effettuare l'allenamento su ciascuno dei modelli considerati. I risultati ottenuti dai diversi regressori sono mostrati nella tabella sottostante e nella figura 50.

Modello	Explained_variance	R2	MAE	MSE	RMSE
LinearRegression	0.9293	0.9293	0.0247	0.0011	0.0337
SVR	0.8703	0.8702	0.0357	0.0021	0.0457
GradientBoostingRegressor	0.7677	0.767	0.0479	0.0037	0.0612
AdaBoostRegressor	0.8878	0.8875	0.0321	0.0018	0.0425
RandomForestRegressor	0.9424	0.9424	0.0217	0.0009	0.0305

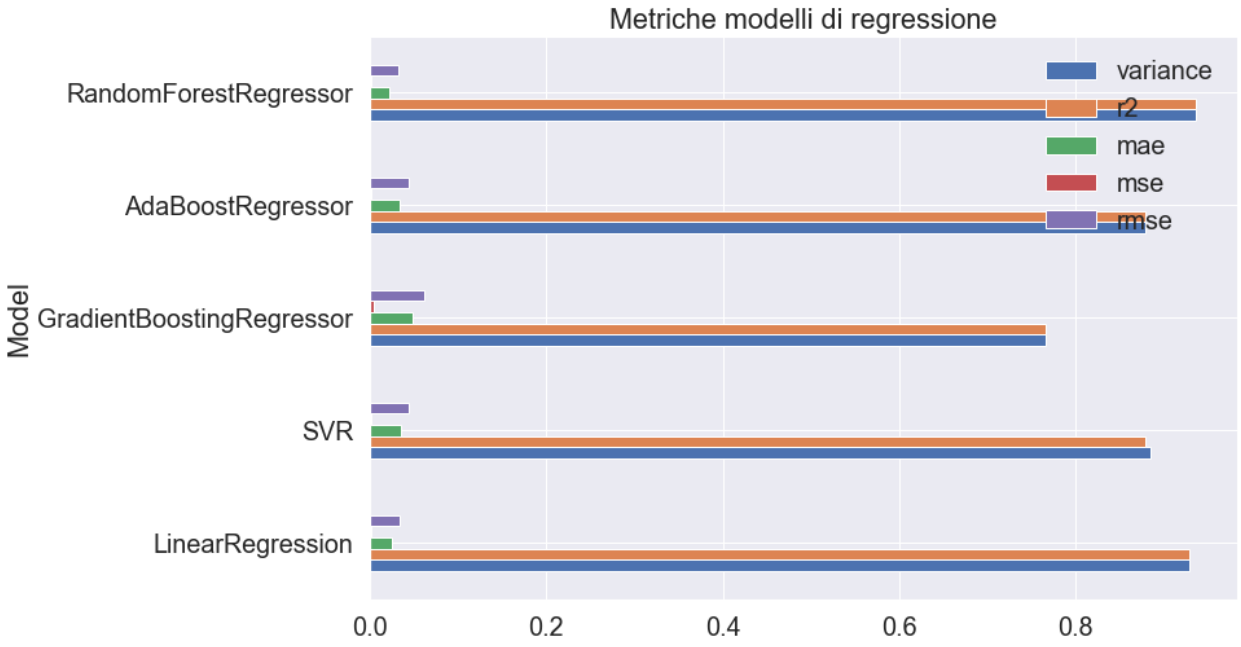


Figura 50: Risultato ottenuto dai diversi regressori

È possibile notare come i modelli più accurati sono il Random Forest e il Linear Regressor, i quali presentano i valori più bassi dell'RMSE sia sul training set che sul test set, e un indice R-squared prossimo a 1. Dall'altro lato, invece, si hanno delle prestazioni mediamente inferiori per l'SVR e l'Adaboost Regressor. Mentre per quanto riguarda il Gradient Boosting Regressor si hanno delle prestazioni leggermente inferiori rispetto a tutti gli altri per il task di regressione.