

# Dengue prediction project

## #grupa 1

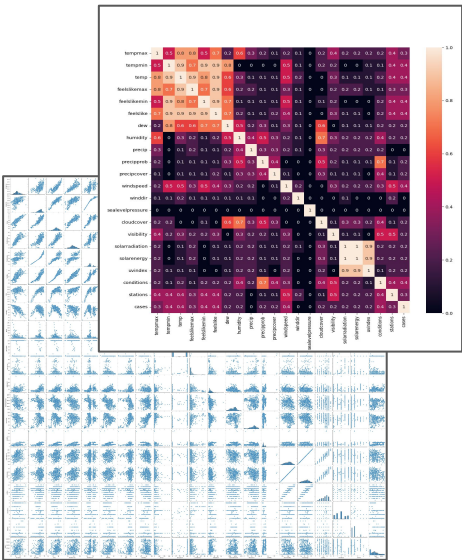
Analiza zbioru danych dotyczących ilości przypadków zachorowań na gorączkę Denga



# Dane wejściowe - przygotowanie danych do analizy (EDA)

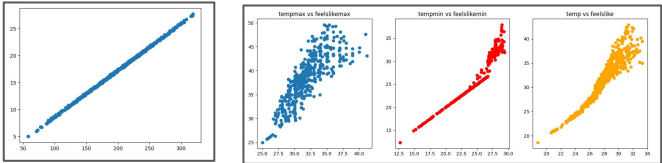
Dostarczone dane zawierają 602 wiersze i 24 kolumny danych z warunkami pogodowymi oraz ilością przypadków zachorowań 'cases' jako zmienna zależna.

- Zmienne niezależne
- tempmax,
  - tempmin,
  - temp,
  - feelslikemax,
  - feelslike,
  - dew,
  - humidity,
  - precip,
  - precipprob
  - precipcover
  - snow
  - snowdepth
  - windspeed
  - winddir
  - sealevelpressure
  - cloudcover
  - visibility
  - solarradiation
  - solarenergy
  - uvindex
  - conditions
  - stations
  - labels
- Zmienne zależne
- cases



Wykres pairplot - pokazujący zależności pomiędzy danymi. Jest on mało czytelny, jednak wzrokowo można wytypować z niego korelacje pomiędzy niektórymi danymi i następnie zająć się ich obróbką

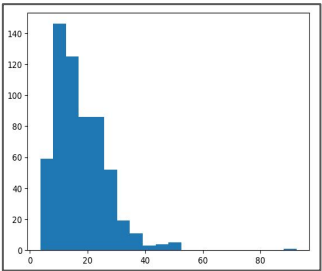
Dane nieadekwatne do analizy takie jak snow, snowdepth, winddir zostały usunięte ze zbioru, jako niemające wpływu na zana zależna - ilość zachorowań 'cases'.



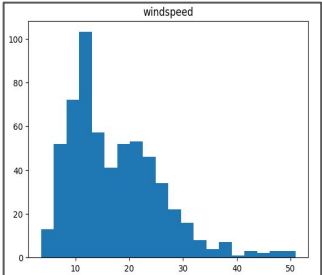
Przypadek solarradiation i solarenergy - sa ze sobą ściśle skorelowane i jedna z nich została usunięta

Pozostałe dane niezależne zostały poddane działaniom mającym na celu usunięcie wartości odstających albo nierealistycznych. Do celu tego posłużyły histogramy i arbitralne decyzje o pozostawieniu lub usunięciu wierszy. Najczęściej były usuwane wartości powyżej 99-ego percentyla (po kilka sztuk lub pojedyncze wartości).

Rozkład wartości windspeed przed obróbką



Rozkład po odcięciu wartości > 99 percentyla



Po przeprowadzeniu EDA zbiór zawiera 592 wiersze i 17 kolumn

# Regresja liniowa

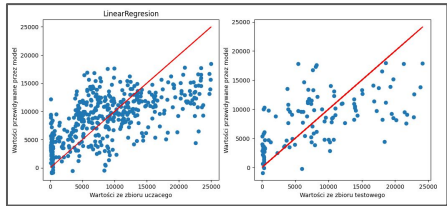
Celem poniższych działań jest stworzenie modelu predykcyjnego. Model ma za zadanie przewidzieć **ilość zachorowań** na podstawie **danych meteorologicznych i środowiskowych**.

Dane z poprzedniej analizy EDA zostały podzielone na dwa zestawy treningowy train oraz test w stosunku 80/20%.

**X** - zbiór danych niezależnych (meteorologiczne i środowiskowe)

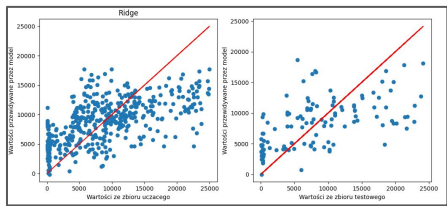
**y** - zbiór danych zależnych (ilość zachorowań)

Przed stworzeniem modeli regresji dane dodatkowo zostały poddane procesowi standaryzacji mającej na celu usprawnienie procesu uczenia modeli ML.



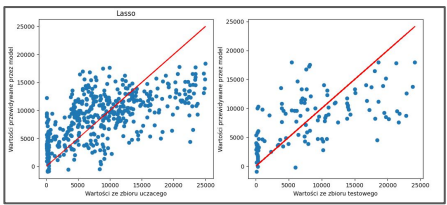
## Model LinearRegression

<b>TRAIN:</b>	<b>TEST:</b>
MAE: 4210.929	MAE: 4498.861
RMSE: 5221.816,	RMSE: 5672.107
R2: 0.405	R2: 0.316



## Model Ridge

<b>TRAIN:</b>	<b>TEST:</b>
MAE: 4282.884	MAE: 4587.643
RMSE: 5277.084	RMSE: 5690.729
R2: 0.393	R2: 0.312



## Model Lasso

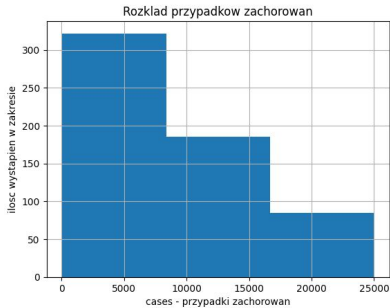
<b>TRAIN:</b>	<b>TEST:</b>
MAE: 4212.36	MAE: 4497.805
RMSE: 5221.963	RMSE: 5670.3
R2: 0.405	R2: 0.317

Do oceny sprawności działania modeli zostały użyte następujące metryki:  
MAE - mean square error  
RMSE - root mean square error  
R2 - coefficient of determination

Zasadniczo wszystkie modele wykazują się porównywalnym poziomem sprawności. Przy czym model Ridge przewiduje wartość ilości zachorowań nieznacznie gorzej od LinearRegression i Lasso. Zasada ta obowiązuje zarówno danych ze zbioru uczącego jak i testowego.

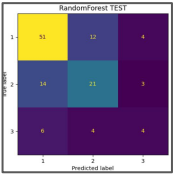
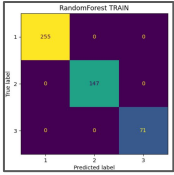
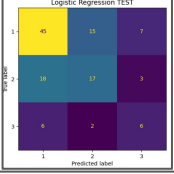
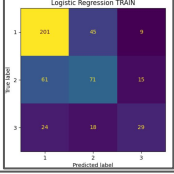
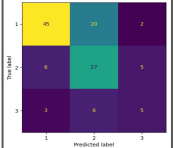
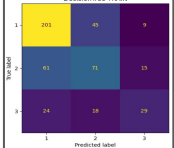
# Modele klasyfikacyjne

Ze względu na fakt że dane wejściowe nie posiadały żadnych cech kategorycznych które można byłoby wykorzystać jako klasę (etykietę / label) do predykcyjnego modelu klasyfikacyjnego. Utworzona została dodatkowa kolumna bazująca na ilości przypadków zachorowań. Ilość przypadków zachorowań została podzielona na 3 przedziały wynikające z rozkładu widocznego na histogramie z 3-ma wartościami. Kolumna z danymi dotyczącymi ilości przypadków została usunięta ze zbioru.



label	ilość przypadków	zachorowalność
1	poniżej 8362	niska
2	8362 - 16672	średnia
3	powyżej 16672	wysoka

Dane zostały poddane standaryzacji.

TEST	TRAIN	
		RandomForest
		Logistic Regression
		DecisionTree

	RandomForest	LogisticRegression	DecisionTreeClassifier
accuracy_test	0.630	0.571	0.655
accuracy_train	1.000	0.636	0.778
f1_test	0.549	0.511	0.578
f1_train	1.000	0.572	0.760
precision_test	0.556	0.509	0.581
precision_train	1.000	0.593	0.780
recall_test	0.543	0.516	0.585
recall_train	1.000	0.560	0.769

Model **DecisionTreeClasifier** (drzewo decyzyjne) wykazał się najlepszymi zdolnościami poprawnego zakwalifikowania zmiennych niezależnych i wytypowania odpowiedniej przynależności do klasy. Wyższą skuteczność modelu uzyskana została po dostosowaniu hyperparametru max\_depth=7 ograniczająca 'głębokość gałęzi decyzyjnych'. Pozwoliło to również zapobiec nadmiernemu dopasowaniu modelu jakim się charakteryzował bez tego parametru.

Metryki modelu **RandomForest** (Las losowy) przyjmują wartość 1.0 co oznacza nadmierne dopasowanie modelu (przetrenowanie).

Model **LogisticRegression** charakteryzuje się wynikami wyraźnie gorszymi w porównaniu z modelem DecisionTree

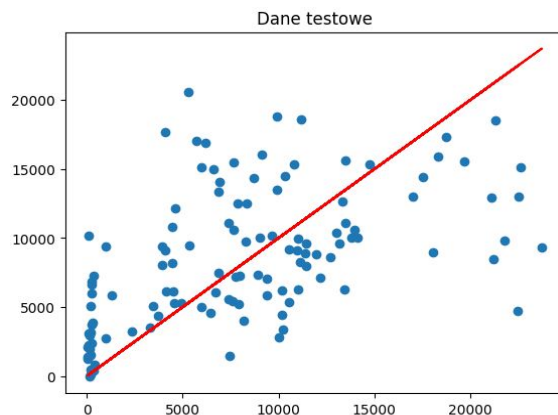
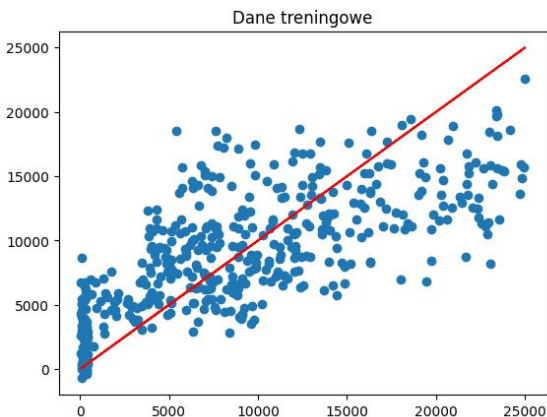
# Model ANN - Artificial Neural Network z wykorzystaniem biblioteki TensorFlow / keras

Do budowy modelu uczenia głębokiego DL została użyta sieć neuronowa składająca się z 2 warstw + warstwa wyjściowa.

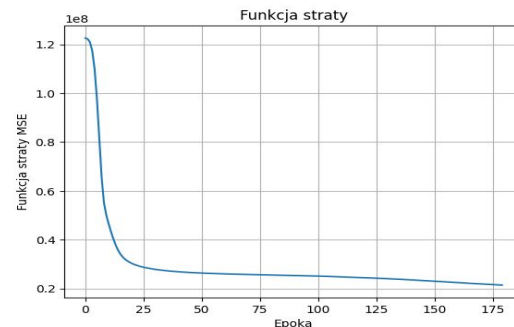
Do treningu modelu zostały użyte dane po ówczesnej standaryzacji.

## Hyperparametry modelu

- Warstwa (1) - **128** neuronów, funkcja aktywacyjna **Leaky\_Relu**
- Warstwa (2) - **32** neurony, funkcja aktywacyjna **Relu**
- Warstwa (wyjściowa) - **1** neuron
- Algorytm optymalizacyjny - **Adam**
- Learning rate = **0.005**
- Ilość epok = **180**
- Funkcja straty loss = **mean\_squared\_error**



Parametry modelu takie jak: ilość warstw, ilość neuronów oraz ilość epok, były dobierane ręcznie, na zasadzie obserwacji a metryka użyta do oceny jego sprawności to R2\_SCORE



R2\_SCORE TEST: 0.211

R2\_SCORE TRAIN: 0.559

Zwiększenie liczby epok, nie przynosiło poprawy sprawności modelu dla danych testowych, a jedynie dla danych treningowych, co w zasadzie prowadziło jedynie do przetrenowania modelu

# Dyskusja wyników i wnioski końcowe

## Regresja:

Z porównania sprawności modeli regresyjnych najkorzystniejsze wartości metryki `R2_SCORE` wykazały się modele regresji liniowej z wartościami **0.31**. Znacznie gorszymi wynikami wykazał się model ANN oparty o 3 warstwową sieć neuronową `R2_SCORE = 0.21`. Warty zaznaczenia jest fakt że wyniki te były mocno zależne od wartości parametru `random_seed` służącej do inicjalizacji generatora liczb losowych przy podziale danych na zbiór treningowy i testowy. Oznaczać to może iż zbiór danych jest zbyt mały do stabilnego wytrenowania modeli oraz słabe zbalansowanie różnorodnych wartości zmiennej zależnej w tych zbiorach. Próby zmiany podziału z domyślnie użytego stosunku 80/20 na 70/30 i nawet 50/50 nie przynosiły poprawy sprawności modeli.

## Klasyfikacja:

Potencjalnym sposobem na poprawę sprawności działania modeli klasyfikacyjnych jest zbalansowanie danych, w chwili obecnej w klasie '3' odpowiadającej wysokiej liczbie zachorowań jest niewiele danych. Pomocnym również mogłaby okazać się zaawansowany dobór hyperparametrow modeli.

## Zalecenia:

Użycie technik balansowania przy podziale danych na zbiór treningowy i testowy w celu uniknięcia niedoprobkowania i nadpróbkowania.

Użycie zaawansowanych technik do strojenia hyperparametrow modeli.

## Konkluzja:

Najlepsze rezultaty uzyskane zostały przy użyciu następujących modeli:

Regresja: Model Lasso `R2=0.31`

Klasyfikacja: `DecisionTreeClassifier` `accuracy=0.67`