ADS Project 1: The Deluge - Flood Risk in the UK

Presented by Team Trent

Objectives and project scope

Project Deluge had the objective to predict and map flood risk across the UK, as well as provide access to relevant live weather data. The aim was to create a python package that a)estimates and validates the best models, b) provides interactive visual representation of the data and c) provide interactive user interface.

Methodology

In order to predict the three target features (flood risk, median house price, local authority) different ML models were evaluated on the training dataset using GridSearch. The primary input models were selected based on the nature of the task (regression or classification), and further narrowed down based on their scoring (negative MSE) relative to their training time. The individual model parameters were tuned in the same manner in order to obtain a ranking of models based on performance. Training features were transformed to numerical objects and selected based on the target and uniqueness of the features. Learning curves were consulted to prevent overfitting.The important features for regression are Easting, Northing, Altitude for numerical and Sector and SoilType for Categorical. The rest of the features are not used as it not related with the targets. For the visualisation tool to show weather data, it was mapped via the folium package, and a webscrapping tool was built to obtain live station information.

Results

The best performing models for the regression tasks (flood risk and median house price estimations) are shown below. Models were stored as methods for flexible use in the tool. Due to the nature of the local authority estimation, a KNN classifier with n_neighbors = 1 was selected here to point to the nearest available sample by euclidean distance, with the only 'training' input being the coordinate features.

	negative_rmse	training_time	regressor
0	-3830844.944433	4.569261	SGDRegressor(max_iter=5000)
2	-4030118.722792	0.037264	KNeighborsRegressor()
5	-4314998.886764	36.382027	RandomForestRegressor()
3	-4939622.441208	0.284685	ElasticNet()
1	-5229886.405503	1.389546	GradientBoostingRegressor()
4	-5258531.633483	0.315447	LinearRegression()
6	-5577641.924968	0.815622	DecisionTreeRegressor()

	neg_mean_squared_error	mean_train_time	Regression Model
1	-0.262401	1.883393	GradientBoostingRegressor()
2	-0.266679	2.833946	SVR()
4	-0.269983	0.033641	KNeighborsRegressor()
0	-0.276908	0.064427	SGDRegressor(max_iter=5000)
6	-0.289839	1.132631	LinearRegression()
3	-0.298672	0.081725	ElasticNet()
5	NaN	0.064269	LogisticRegression()

Figure 1: Result of GridSearch for best regression model for median house price (left) and flood risk (right) prediction. Only SGDRegressor was used to estimate median house price, KNN Regressor did not converge and RandomForestRegressor took too much time to train. For the flood risk prediction, 2 models were selected as methods that the user can decide between.

An example of the data visualization is shown below, using an example of a day with typical levels of precipitation.

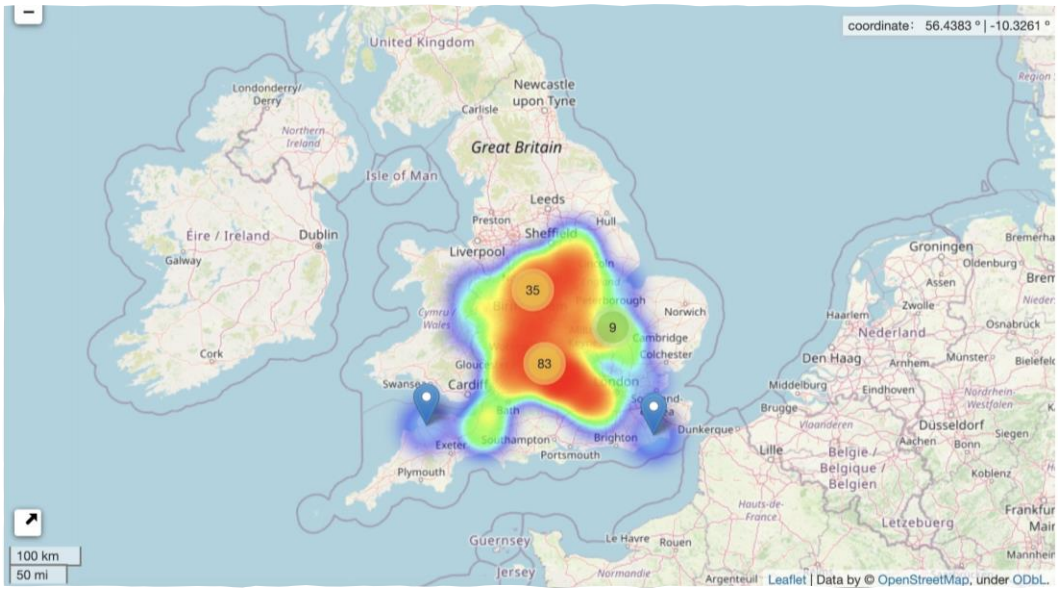

Figure 2: 'At risk' river and tide levels mapped across the UK. Data is looked up live from environment.data.gov.uk

Discussion & Conclusion

The mean absolute error for risk probability prediction is too large to identify classes 1, 2 and 3. However, it performs well in predicting high risks which is most important. While the model scores and rankings help us in the process of choosing methods, it is important to note that the end-quality of the predictions is highly dependent on the data input. However, the development process was limited by time and computational power, and improvements to the modelling process should be investigated in the future. These could include more extensive hyperparameter tuning, the inclusion of additional data, or feature selection and engineering in the current training set.

Figure 3: Trent development team