Assignment 6 Summary

Part 1 – EDA

- Basic EDA – we've looked at the different features and how they correlate with our target variable – mainly boxplots. Based on the results we've seen and on the percentage of missing values in the features (and on some common sense of course), we decided to drop some columns and chose the features we thought will be relevant (e.g. "brand", "category", "occasion" and more).
- We also saw that there are some features that might contain interesting information that we can extract from them, like title, seller notes and color.

Part 2 – cleaning and preprocessing!

We put a lot of work into cleaning and preprocessing. There's not enough space in one page to describe everything, so these are the main points:

- Brand cleaning: First, we used the title feature to extract the brand for datapoints with missing brands – It worked pretty well.
- After trying several methods, we decided to divide the brands to several bins based on their popularity and their average price – the most popular brands got their own category, and the others were put in a "bin" with other brands that have similar average price. (We later created additional features that hold information about the brands "priciness" that were even more helpful, see notebook).
- Additionally, we processed features that had many categorical values ("occasion", "material", "heel_type…), and based on common sense or quick google search we aggregated several categories together and put the less frequent ones as "other".
- We used regexes to extract the heel height (and its unit!), and then converted it to one of 6 categories, but it actually wasn't a very important feature after all ☹.
- Converted categorical features into dummy variables – right before training model.

Part 3 – Fitting different models

- We tried 5 models: Random Forest Regression, Gradient Boosting Regression and lightgbm's LGBMRegressor, which is also GB model, and KNN. After we achieved a good enough score with RF, we also tried NN and actually got better results with mush less work…but as most of our work wasn't centered around this model, we'll write more about it in the model's notebook.
- After splitting the whole train data to train and test sets, we ran randomized search cross validation to decide which are the best hyper parameters for each model and compare the models using the data from the check set. We mainly used random forest (our final chosen model) and gradient bosting, but also tried to run KNN.
- After a few times we focused only on Random Forest though, because of time constraints and because we understood that feature engineering is much more important at this stage.

Part 4 – Evaluating the model & back to stages 1-3

After each submission we went back to stages 1 - 3. We tried to understand how we can improve our model by looking at our prediction error by different features. For example, we saw that we don't predict well high prices even though the brand is known, so we added a numeric parameter of the mean price of expensive brands only (according to the train data only of course), and it really improved our model.