# D1: KAGGLE-HOUSE PRICES

Predict sales prices and practice feature engineering, RFs, and gradient boosting
https://www.kaggle.com/c/house-prices-advanced-regression-techniques

Team: Sandra Lannes, Sarah Lannes, Danver Hans Värv

Project repository: https://github.com/sandralannes/DSproject_Danverjakaksikud

## Business understanding

### Identifying your business goals

#### Background

For our project, we wanted to take the project offered by Bolt. As a result of our e-mail probably getting lost and no answer from Bolt's contact person, we decided to choose a Kaggle competition on the topic of predicting the house prices. In the competition description, it was said to be a great competition for data science students who have completed a machine learning course and are looking to expand their skill set before trying a featured competition. It seemed like a great match with our skill level, so we joined the competition.

In the competition, we are given a dataset with a number of different variables describing almost every aspect of residential homes in Ames, Iowa. The Ames Housing dataset was compiled by Dean De Cock, who teaches a regression course. Dean's main goal was to compose a dataset that lets students use all the knowledge they have learned from their data science course. The ideal data set needed to have a reasonably large number of variables and observations so that students would have to go beyond a simple algorithm, such as forward or stepwise selection, to construct a final model. So he compiled a dataset that satisfied his needs and criterias for the perfect study material.

#### Business goals

The main goal for this project is just to learn and consolidate our knowledge on data science. In addition, with this project we will practice creative feature engineering skills and advanced regression techniques like random forest and gradient boosting. Goal is to predict house prices as accurately as possible using everything we have learned so far.

### Business success criteria

We would like to be in the top 50% of the most accurate teams. Our success is evaluated on Root-Mean-Squared-Error.

## Assessing your situation

### Inventory of resources

For our project we will use our knowledge learned during our “Introduction to Data Science” course, and if needed we will seek help from the course instructors or from the internet (kaggle competition forum, google, youtube, etc.). The data we will be using to complete our project also comes from kaggle, where we have a training dataset and test dataset. We also have a text file which describes the columns in our data and we also have a sample submissions file, for correct submissions.  For our project we will use our computers, which we got from University of Tartu Institute of Computer Science, and to evaluate the results we will use programming language Python with Jupyter and to share the code we will use a Github repository.

### Requirements, assumptions, and constraints

Most of the requirements for our project have been brought out under the rules of the Kaggle competitions. It has been said that it is allowed to have one account per participant and there is no maximum team size. Privately sharing code or data outside of teams is not permitted, although it's okay to share and use code if made available to all participants on the forums. It has been asked that teams respect the spirit of the competition and do not cheat. Hand-labeling is also forbidden. It is possible to submit a maximum of 5 entries per day and teams may select up to 2 final submissions for judging. There is no end date to the competition, however we have to submit our group project for the course “Introduction to Data Science” before Thursday, December 17, 2020.

### Risks and contingencies

We are going to do this project from home. So we rely only on the internet connection. In case of losing connection with the wifi, each member can provide the internet with a hotspot.

### Terminology

We found that all of the business and data-mining terms that we have used so far are familiar to us, since we have participated carefully in all of the lectures and practicals.

### Costs and benefits

As the kaggle competition is free and we have all the needed inventory, we have no costs. As we are doing this project just for ourselves and not for any business, we also don’t have any benefits with this project.

## Defining your data-mining goals

### Data-mining goals

Our data-mining goal is to find out which parameters affect the price the most and therefore construct a good model to predict most accurate prices for houses. At the end of the course, we are going to make a poster and present our results to the students taking this course.

### Data-mining success criteria

Results are evaluated on Root-Mean-Squared-Error between the logarithm of the predicted value and the logarithm of the observed sales price. Taking logarithms means that errors in predicting expensive houses and cheap houses will affect the result equally. We would like to end up in the first half of the leaderboard, so we should have our RMSE difference under 0.14.

# Data understanding

## Gathering data

### Outline data requirements

We need a dataset for training our models, test dataset for testing our models, data description file for knowing which type of data is in each column and sample submission file to know in which format we need to submit our results.

### Verify data availability

All the needed data is available and listed under Kaggle competition.

### Define selection criteria

For training the model we will use a train.csv file. While exploring our data, we will find the key fields that affect the price and will rely on them the most.

## Describing data

Our data is provided by Kaggle and given in comma-separated values file format. We will be using train.csv to train a model that will predict house prices based on the given data, and then we will test our model on test.csv. We have in total 81 columns in our training set, from which 48 are String type, 30 are integer type, 1 is ID and 1 is other type. There are 1460 cases in training dataset and 1459 cases in testing dataset. Descriptions of the columns were listed in the given data descriptions file and have been brought out in the list below. The training file has enough cases and columns for quality evaluations and results.

## Description of columns

- MSSubClass: Identifies the type of dwelling involved in the sale.
- MSZoning: Identifies the general zoning classification of the sale.
- LotFrontage: Linear feet of street connected to property
- LotArea: Lot size in square feet
- Street: Type of road access to property
- Alley: Type of alley access to property
- LotShape: General shape of property
- LandContour: Flatness of the property
- LotConfig: Lot configuration
- LandSlope: Slope of property
- Neighborhood: Physical locations within Ames city limits
- Condition1: Proximity to various conditions
- Condition2: Proximity to various conditions (if more than one is present)
- BldgType: Type of dwelling
- HouseStyle: Style of dwelling
- OverallQual: Rates the overall material and finish of the house
- OverallCond: Rates the overall condition of the house
- YearBuilt: Original construction date
- YearRemodAdd: Remodel date (same as construction date if no remodeling or additions)
- RoofStyle: Type of roof
- RoofMatl: Roof material
- Exterior1st: Exterior covering on house
- Exterior2nd: Exterior covering on house (if more than one material)
- MasVnrType: Masonry veneer type
- MasVnrArea: Masonry veneer area in square feet
- ExterQual: Evaluates the quality of the material on the exterior

- ExterCond: Evaluates the present condition of the material on the exterior
- Foundation: Type of foundation
- BsmtQual: Evaluates the height of the basement
- BsmtCond: Evaluates the general condition of the basement
- BsmtExposure: Refers to walkout or garden level walls
- BsmtFinType1: Rating of basement finished area
- BsmtFinSF1: Type 1 finished square feet
- BsmtFinType2: Rating of basement finished area (if multiple types)
- BsmtFinSF2: Type 2 finished square feet
- BsmtUnfSF: Unfinished square feet of basement area
- TotalBsmtSF: Total square feet of basement area
- Heating: Type of heating
- HeatingQC: Heating quality and condition
- CentralAir: Central air conditioning
- Electrical: Electrical system
- 1stFlrSF: First Floor square feet
- 2ndFlrSF: Second floor square feet
- LowQualFinSF: Low quality finished square feet (all floors)
- GrLivArea: Above grade (ground) living area square feet
- BsmtFullBath: Basement full bathrooms
- BsmtHalfBath: Basement half bathrooms
- FullBath: Full bathrooms above grade
- HalfBath: Half baths above grade
- Bedroom: Bedrooms above grade (does NOT include basement bedrooms)
- Kitchen: Kitchens above grade
- KitchenQual: Kitchen quality
- TotRmsAbvGrd: Total rooms above grade (does not include bathrooms)
- Functional: Home functionality (Assume typical unless deductions are warranted)
- FireplaceQu: Fireplace quality
- GarageType: Garage location
- GarageYrBlt: Year garage was built
- GarageFinish: Interior finish of the garage
- GarageCars: Size of garage in car capacity
- GarageArea: Size of garage in square feet
- GarageQual: Garage quality
- GarageCond: Garage condition
- PavedDrive: Paved driveway

- WoodDeckSF: Wood deck area in square feet
- OpenPorchSF: Open porch area in square feet
- EnclosedPorch: Enclosed porch area in square feet
- 3SsnPorch: Three season porch area in square feet
- ScreenPorch: Screen porch area in square feet
- PoolArea: Pool area in square feet
- PoolQC: Pool quality
- Fence: Fence quality
- MiscFeature: Miscellaneous feature not covered in other categories
- MiscVal: $Value of miscellaneous feature
- MoSold: Month Sold (MM)
- YrSold: Year Sold (YYYY)
- SaleType: Type of sale
- SaleCondition: Condition of sale

### Exploring data

Each column has correct values. When working with our dataset, we probably need to assign numeric values to all columns that don’t have numeric values.

### Verifying data quality

Overall, we have a high quality dataset. The dataset has a lot of NA values, but in our case, the values are not missing, these data objects just don’t have those things(houses without pools, alleys etc).

# Planning your project

Make a detailed plan of your project with a list of tasks. There should be at least 5 tasks. Specify how many hours each team member is going to contribute to each task.

List the methods and tools that you plan to use. Add any comments about the tasks that you think are important to clarify.

1. Making the project plan ~3h each member
2. Import dependencies, sklearn etc.
3. Data understanding -  we will make an exploratory analysis of the dataset and provide some observations ~2 h each team member

4. Exploratory data analysis - scatter plots and histograms. We have to find out key elements that affect the price, for this we will use scattering. For example location vs price, bedrooms vs price, etc. We will use seaborn for this task. Also geopandas to plot data on the state map. ~3h each team member
5. Replace missing values, where necessary, avoid replacing N/A whether the house has a pool or not etc. ~1h in total
6. One-Hot Encode our data, set all string values to 1 and 0. ~1h in total
7. Split data into train and test randomly, for finding optimal solutions. ~1h each member
8. Train a machine learning model - train several models with the goal of finding the best model that fits our data. Linear regression, Decision Trees, Random Forest, Support Vector Machine. Exploring new models that we haven't learned yet. ~7h each member
9. Evaluating our models on the test datasets ~4h each member
10. Making a project poster and presenting it in the poster session. ~5h each member