Predicting Future Sales

Team members: Fedor Stomakhin, Egert Metsandi, Rain Trubetski

Business understanding

Identifying our business goals

Background

Our client, one of the largest Russian software firms – 1C Company, seeks to predict total sales for every product and store in the next month by working with a challenging time-series dataset consisting of daily sales data. By solving this task correctly, we will be able to enhance and apply our data science skills.

Business goals

- Increase profits by stocking up more on popular items and less on scarcely bought items by month

Business success criteria

The business becomes more profitable as a result of our modeling.

Assessing our situation

Inventory of resources

Files: Sales_train.csv.gz, test.csv.gz, sample_submission.csv.gz, items.csv, item_categories.csv, shops.csv. For workforce we have three computer science students.

Requirements, assumptions, and constraints

- Requirements
  - Create a working prediction model by December 19.
- Assumptions
  - Data for the task is available
- Constraints
  - The list of shops and products changes slightly every month, meaning that the older the data on which the model is trained, the less accurate it is. The model has to be able to handle a situation in which test data contains previously unknown products and shops.

Risks and contingencies

If there's a power outage or no internet connection we will go to the public library.

If the project takes longer than anticipated we will speed it up.

Terminology

- sales_train.csv - the training set. Daily historical data from January 2013 to October 2015.
- test.csv - the test set. We need to forecast the sales for these shops and products for November 2015.
- sample_submission.csv - a sample submission file in the correct format.
- items.csv - supplemental information about the items/products.
- item_categories.csv - supplemental information about the items categories.
- shops.csv- supplemental information about the shops.

- ID - an Id that represents a (Shop, Item) tuple within the test set
- shop_id - unique identifier of a shop
- item_id - unique identifier of a product
- item_category_id - unique identifier of item category
- item_cnt_day - number of products sold. We are predicting a monthly amount of this measure
- item_price - the current price of an item
- date - date in format dd/mm/yyyy
- date_block_num - a consecutive month number, used for convenience. January 2013 is 0, February 2013 is 1,..., October 2015 is 33
- item_name - name of the item
- shop_name - name of the shop
- item_category_name - name of item category

Costs and benefits

- Costs:
  - A total of 90 hours of work (~289 euros worth of minimum wage work)
  - 12 cans of Red Bull.
  - 12 cans of Pringles.

- Benefits:
  - 20 points in the subject
  - Greater turnover for the company

Defining our data-mining goals

Data-mining goals

- Make the model robust enough to handle the monthly changes in the shops and products
- Predict product sales amount for the next month in every shop
- Predict which item categories sell the best and plot the relations

The success criteria for data mining would be in this instance an understanding of the data that is sophisticated enough to begin preprocessing the train data for the prediction model. This understanding has been partially reached, as can be observed from Terminology. The next step to develop this understanding would be to explore and summarize the relationships between various columns.

Data understanding

Gathering data

Outline data requirements

We require labeled training data with all the columns in sales_train.csv: 'date', 'date_block_num', 'shop_id', 'item_id', 'item_price', 'item_cnt_day', elaborated upon in Terminology. A column with appropriate item category ID-s will be appended.

Verify data availability

Presently we have several separate .csv-s. The base training data does not contain mappings of product ID to product type IDs, rather they are contained in items.csv. The greatest challenge comes with the prediction of the data in the test (test.csv.gz) set, as it does not contain month nor product ID data. However, this can be solved by utilizing the mappings generated from items.csv, for example. In the case that the item is not contained in the items.csv list, a default parameter will be used, e.g "other".

Define selection criteria

We will select training data with columns pertaining to product ID (int), product type ID (int), store ID (int), month (int) and the amount of products sold on that date (int). Months would be counted from January 2013, with it being the 0th month. Product ID and Product type ID would be translated into one-hot vectors.

Describing data

- All of the data besides numerical values and column titles are in Russian
- Training set for daily historical data from January 2013 to October 2015, which contains the amount of different products sold at a certain date in a certain store.
- A test set for November 2015 where we predict the sales.
- Three CSV files with supplemental information about the items, item categories, and shops.
- There is a sample file that serves as an example for submitting our data to kaggle

Exploring data

While the data is ordered by month from 0 to 33 (Jan 2013 to Oct 2015), the exact dates in each month are not in a specific order. There are also 7252 items where item_cnt_day is -1.0, which we assume, with no actual proof, are items that were later returned to shops. As the data only shows items that were sold, there is no item with a value of 0.0. There are a total of 84 different item categories in the dataset. As our aim is to predict sales for a month total, not by day, we can ignore the ‘date’ column and only use date_block_num. There are clear periodic trends for the popularity of some item categories over others, e.g the relationship between various “gifts” (“Подарки”) subcategories and months November, December, and January. We might have to extract a month category from the date_block_num numerical value, as the same pattern can be observed with different stores - a periodic uptick during the holiday season. In addition, it is evident how the sale of some item categories has decreased over time - for example, PSP sales.

Verifying data quality

As we have no proof for what the items with item_cnt_day -1.0 represent, we will not use them. There are no NaNs or null values in the data.

Planning our project

1.	Creating a git repository and a notebook for initial data exploration.	1h	-Egert
2.	Explore data and summarize the findings.	6h	-Fedor,Rain
3.	Clean the data.	5h	-Egert,Rain
4.	Wrangle the data.	10h	-Fedor,Egert
5.	Preprocess the data for input to the model.	13h	-Fedor,Egert
6.	Design a model for predicting sales for the next month	20h	-All
7.	Evaluation of the results and re-training the model	20h	-All
8.	Create poster	15h	-All

Methods and tools we plan to use

- Data wrangling. We intend to aggregate our data at the monthly level and sum up the sales, as the test dataset does not include dates - we are predicting sales for the whole month. In addition, we intend to generate additional columns - to account for variation in sales between different months, and to add item category data to the training set. We will use pandas and numpy to handle the data, and seaborn with matplotlib to review the results of the wrangling.
- Creating a deep learning neural network model. We will use keras and tensorflow to create the model.
- Data visualization. We will used seaborn and matplotlib for this.