How to Rank 10% in Your First Kaggle Competition

<= Previous post Next post =>

Tags: Beginners, Competition, Data Science, Kaggle, Machine Learning, Python

This post presents a pathway to achieving success in Kaggle competitions as a beginner. The path generalizes beyond competitions, however. Read on for insight into succeeding while approaching any data science project.

Linghao Zhang, Fudan University.

Introduction

money to set up data science competitions on Kaggle. Recently I had my first shot on Kaggle and **ranked 98th (~ 5%) among 2125 teams**. Being my Kaggle debut, I feel quite satisfied with the result. Since many Kaggle beginners set 10% as their first goal, I want to share my two cents on how to achieve that.

This post is also available in Chinese.

Updated on Oct 28th, 2016: I made many wording changes and added several updates to this post. Note that Kaggle has went through some major changes since I published this post, especially with its ranking system. Therefore some descriptions here might not apply anymore.

Most Kagglers use Python or R. I prefer Python, but R users should have no difficulty in understanding the ideas behind tools and languages.

First let's go through some facts about Kaggle competitions in case you are not familiar with them.

- Different competitions have different tasks: classifications, regressions, recommendations... Training set and testing set will be open for download after the competition launches.
- A competition typically lasts for 2 ~ 3 months. Each team can submit for a limited number of times per day. Usually it's 5 times a day.
- There will be a 1st submission deadline one week before the end of the competition, after which you cannot merge teams or enter the competition. Therefore **be sure to have at least one valid submission before that.**
- You will get you score immediately after the submission. Different competitions use different scoring metrics, which are explained by the question mark on the leaderboard.
- The score you get is calculated on a subset of testing set, which is commonly referred to as a **Public LB** score. Whereas the final result will use the remaining data in the testing set, which is referred to as a **Private LB** score.
- The score you get by local cross validation is commonly referred to as a **CV** score. Generally speaking, CV scores are more reliable than LB scores.
- Beginners can learn a lot from **Forum** and **Scripts**. Do not hesitate to ask about anything. Kagglers are in general very kind and helpful.

I assume that readers are familiar with basic concepts and models of machine learning. Enjoy reading!

General Approach

In this section, I will walk you through the process of a Kaggle competition.

Data Exploration

What we do at this stage is called **EDA (Exploratory Data Analysis)**, which means analytically exploring data in order to provide some insights for subsequent processing and modeling.

Usually we would load the data using Pandas and make some visualizations to understand the data.

Visualization

For plotting, Matplotlib and Seaborn should suffice.

Some common practices:

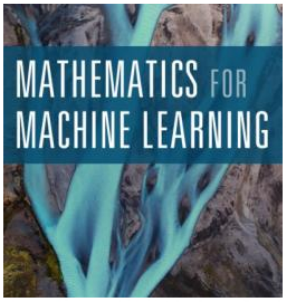
- Inspect the distribution of target variable. Depending on what scoring metric is used,**an imbalanced distribution of target variable might harm the model's performance**.
- For **numerical variables**, use **box plot** and **scatter plot** to inspect their distributions and check for outliers.

Latest News

- 4 Steps to ensure your AI/Machine Learning system survi...
- Dockerize Jupyter with the Visual Debugger
- OpenAI Open Sources Microscope and the Lucid Library to...
- State of the Machine Learning and AI Industry
- Dive Into Deep Learning: The Free eBook
- Better notebooks through CI: automatically testing docu...

Top Stories Last Week

1. Mathematics for Machine Learning: The Free eBook



2. How to Do Hyperparameter Tuning on Any Python Script in 3 Easy Steps
3. COVID-19 Visualized: The power of effective visualizations for pandemic storytelling
4. Uber Open Sourced Fiber, a Framework to Streamline Distributed Computing for Reinforcement Learning Models
5. 24 Best (and Free) Books To Understand Machine Learning
6. How (not) to use Machine Learning for time series forecasting: The sequel
7. How to select rows and columns in Pandas using [ ], .loc, iloc, .at and .iat

Most Shared

1. 10 Must-read Machine Learning Articles (March 2020)
2. Mathematics for Machine Learning: The Free eBook
3. Top KDnuggets tweets, Apr 01-07: How to change global policy on #coronavirus
4. 5 Ways Data Scientists Can Help Respond to COVID-19 and 5 Actions to Avoid
5. How to Do Hyperparameter Tuning on Any Python Script in 3 Easy Steps

- For classification tasks, plot the data with points colored according to their labels. This can help with feature engineering.
- Make pairwise distribution plots and examine their correlations.

Be sure to read this inspiring tutorial of exploratory visualization before you go on.

Statistical Tests

We can perform some statistical tests to confirm our hypotheses. Sometimes we can get enough intuition from visualization, but quantitative results are always good to have. Note that we will always encounter non-i.i.d. data in real world. So we have to be careful about which test to use and how we interpret the findings.

In many competitions public LB scores are not very consistent with local CV scores due to noise or non-i.i.d. distribution. You can use test results to **roughly set a threshold for determining whether an increase of score is due to genuine improvment or randomness**.

Data Preprocessing

In most cases, we need to preprocess the dataset before constructing features. Some common steps are:

- Sometimes several files are provided and we need to join them.
- Deal with **missing data**.
- Deal with **outliers**.
- Encode **categorical variables** if necessary.
- Deal with noise. For example you may have some floats derived from raw figures. The loss of precision during floating-point arithemics can bring much noise into the data: two seemingly different values might be the same before conversion. Sometimes noise harms model and we would want to avoid that.

How we choose to perform preprocessing largely depends on what we learn about the data in the previous stage. In practice, I recommend using **Jupyter Notebook** for data manipulation and mastering usage of frequently used Pandas operations. The advantage is that you get to see the results immediately and are able to modify or rerun code blocks. This also makes it very convenient to share your approach with others. After allreproducible results are very important in data science.

Let's see some examples.

Pages: 1 2 3 4

<= Previous post Next post =>

Top Stories Past 30 Days

Most Popular

1. 24 Best (and Free) Books To Understand Machine Learning
2. COVID-19 Visualized: The power of effective visualizations for pandemic storytelling
3. How (not) to use Machine Learning for time series forecasting: The sequel
4. Mathematics for Machine Learning: The Free eBook
5. Nine lessons learned during my first year as a Data Scientist
6. 50 Must-Read Free Books For Every Data Scientist in 2020
7. How to select rows and columns in Pandas using [ ], .loc, iloc, .at and .iat

Most Shared

1. Introducing MIDAS: A New Baseline for Anomaly Detection in Graphs
2. 24 Best (and Free) Books To Understand Machine Learning
3. COVID-19 Visualized: The power of effective visualizations for pandemic storytelling
4. How (not) to use Machine Learning for time series forecasting: The sequel
5. 10 Must-read Machine Learning Articles (March 2020)
6. Best Free Epidemiology Courses for Data Scientists
7. Coronavirus Data and Poll Analysis – yes, there is hope, if we act now

Easy Steps

6. How Data Science Is Being Used to Understand COVID-19
7. 3 Best Sites to Find Datasets for your Data Science Projects

More Recent Stories

- Better notebooks through CI: automatically testing documentati...
- Top tweets, Apr 08-14: Mathematics for #MachineLearning: Th...
- Pandas in action
- Why and How to Use Dask with Big Data
- Federated Learning: An Introduction
- Visualizing Decision Trees with Python (Scikit-learn, Graphviz...
- KDnuggets 20:n15, Apr 15: How to Do Hyperparameter Tuning o...
- Top Process Mining Software Companies, Updated
- Can Java Be Used for Machine Learning and Data Science?
- Free Metis Corporate Training Series: Intro to Python, Continued
- Forecasting Stories 2: The Power of a Seasonality Index
- Free Workshop Preview: Data Thinking with Martin Szugat
- Peer Reviewing Data Science Projects
- Top Stories, Apr 6-12: Mathematics for Machine Learning: The F...
- How Deep Learning is Accelerating Drug Discovery in Pharmaceut...
- DeepMind Unveils Agent57, the First AI Agents that Outperforms...
- KNIME Spring Summit Online Edition
- Upcoming Webinars and online events in AI, Data Science, Machi...
- Has AI Come Full Circle? A data science journey, or why I acc...
- Successful Use Cases of Artificial Intelligence for Businesses

KDnuggets Home » News » 2016 » Nov » Tutorials, Overviews » How to Rank 10% in Your First Kaggle Competition (16:n41)