Feature Engineering **Meiling Liu**
May 5, 2019

1. 8921483 records in train, 7853253 records in test;
2. 82 feature in total and they can be clustered into following categories
   - don't need engineering;
   - lots of categories, but can extract information from it, merge some categories:
     e.g. EngineVersion: 1.1.15100.1, 1.1.14600.4, etc.
     can be split into 4 sub-features by "." and treat each sub-feature differently.
   - lots of categories, unable to extract information from it, but features has meaning, e.g. country code, etc. If we use them directly, the data might be too noisy and contain irrelevant information. We could convert the original feature categories by using techniques such as **Greedy TS**[1] or **binary encoding**.
   - Remove the **collinearity** within features, such as features "Census_OSEdition" and "Census_OSSkuName" are strongly positively correlation, feature "Census_OSEdition" is removed from the dataset. Similarly, features "Census_OSArchitecture", "Processor", "OsBuildLab", "Census_OSBranch", "Census_OSBuildNumber" , and "Census_OSBuildRevision" are correlation since their information are nested.
3. After the first level processing, we end up with 249 features in total.
4. Perform automated feature engineering by using deep feature synthesis tools *featuretools* in python. The result will provide us a good insight of the features.
5. Apply limma and random forest algorithms and select features that have high importance score in one or both algorithms.

References

[1] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features. In *Advances in Neural Information Processing Systems*, pages 6638–6648, 2018.