基于机器学习的预测模型的建立和实现：

主要分为Data Cleaning，Features Engineering， Models Training

Data Cleaning

1. 移除多余的duplicate features（相同或极为相似的features）

2. 移除constant features（只有一个value的feature）

```
#R里面可以使用unique()函数判断，如果返回值为1，则意味着为constant features
```

3. 移除方差过小的features（方差过小意味着提供信息很有限）

```
#R中可以使用caret包里的nearZeroVar()函数

#Python里可以使用sklearn包里的VarianceThreshold()函数
```

4. 缺失值处理：将missing value重新编为一类。

```
#比如原本-1代表negative，1代表positive，那么missing value就可以全部标记为0

#对于多分类的features做法也类似二分类的做法

#对于numeric values,可以用很大或很小的值代表missing value比如-99999.
```

5. 填补缺失值

可以用mean，median或者most frequent value进行填补

```
#R用Hmisc包中的impute()函数

#Python用sklearn中的Imputer()函数
```

6. 高级的缺失值填补方法

利用其他column的features来填补这个column的缺失值（比如做回归）

```
#R里面可以用mice包，有很多方法可供选择
```

注意：不是任何时候填补缺失值都会对最后的模型预测效果带来正的效果，必须进行一定的检验。

Features Engineering

1. Data Transformation

a. Scaling and Standardization

```
#标准化，R用scale(), Python用StandardScaler()

#注意：Tree based模型无需做标准化
```

b. Responses Transformation

#当 responses 展现 skewed distribution 时候用,使得 residual 接近 normal distribution

#可以用 log(x),log(x+1),sqrt(x)等

2. Features Encoding

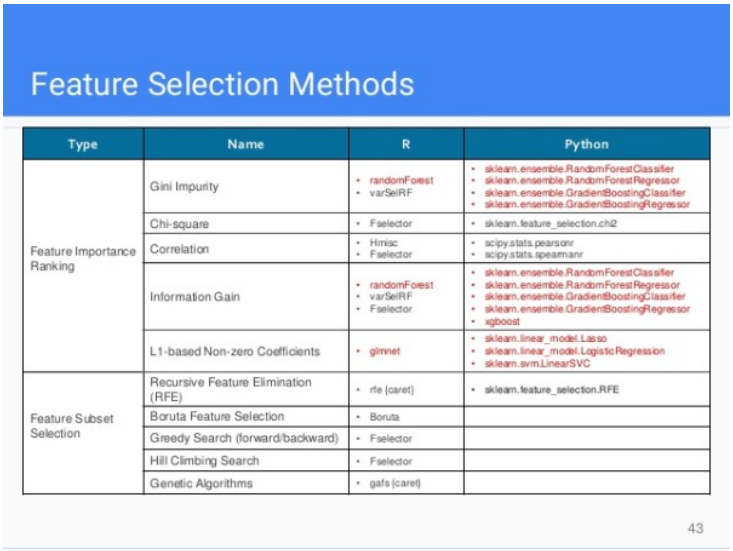
#把 categorical features 变成 numeric feature

#Label encoding：Python 用 LabelEncoder()和 OneHotEncoder(), R 用 dummyVars()

3. Features Extraction

4. Features Selection

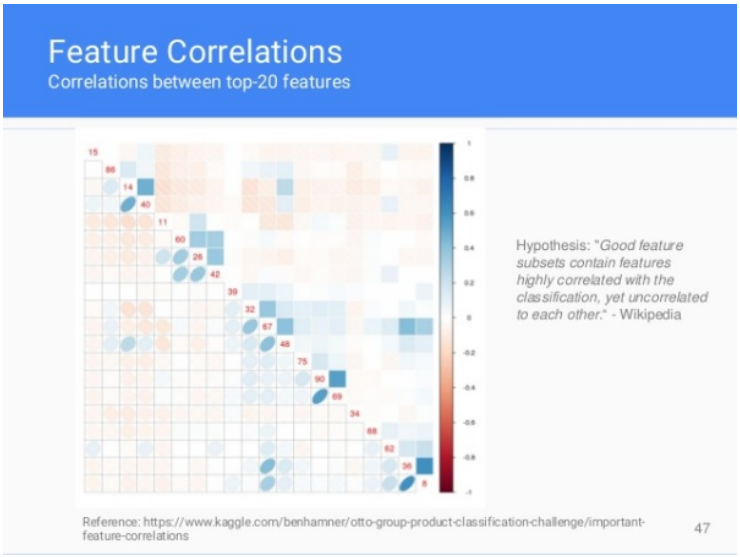
a. 方法很多：



注：其中 randomForest 以及 xgboost 里的方法可以判断 features 的 Importance

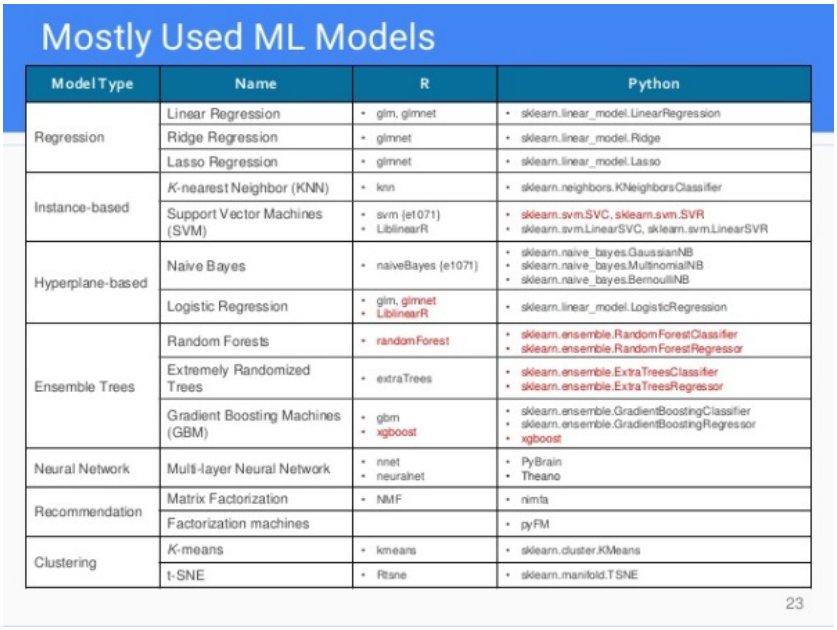
b. 此外，PCA 等方法可以生成指定数量的新 features（映射）

c. 擅对 features 进行 visualization 或 correlation 的分析。



Models Trainning

Mostly Used ML Models



Mostly Used ML Models

Model Type	Name	R	Python
Regression	Linear Regression	• glm, glmnet	• sklearn.linear_model.LinearRegression
	Ridge Regression	• glmnet	• sklearn.linear_model.Ridge
	Lasso Regression	• glmnet	• sklearn.linear_model.Lasso
Instance-based	*K*-nearest Neighbor (KNN)	• knn	• sklearn.neighbors.KNeighborsClassifier
	Support Vector Machines (SVM)	• svm (e1071) • LiblinearR	• sklearn.svm.SVC, sklearn.svm.SVR • sklearn.svm.LinearSVC, sklearn.svm.LinearSVR
Hyperplane-based	Naive Bayes	• naiveBayes (e1071)	• sklearn.naive_bayes.GaussianNB • sklearn.naive_bayes.MultinomialNB • sklearn.naive_bayes.BernoulliNB
	Logistic Regression	• glm, glmnet • LiblinearR	• sklearn.linear_model.LogisticRegression
Ensemble Trees	Random Forests	• randomForest	• sklearn.ensemble.RandomForestClassifier • sklearn.ensemble.RandomForestRegressor
	Extremely Randomized Trees	• extraTrees	• sklearn.ensemble.ExtraTreesClassifier • sklearn.ensemble.ExtraTreesRegressor
	Gradient Boosting Machines (GBM)	• gbm • xgboost	• sklearn.ensemble.GradientBoostingClassifier • sklearn.ensemble.GradientBoostingRegressor • xgboost
Neural Network	Multi-layer Neural Network	• nnet • neuralnet	• PyBrain • Theano
Recommendation	Matrix Factorization	• NMF	• nimfa
	Factorization machines		• pyFM
Clustering	*K*-means	• kmeans	• sklearn.cluster.KMeans
	t-SNE	• Rtsne	• sklearn.manifold.TSNE

23

机器学习的工作流程

①选择数据：将你的数据分成三组：训练数据、验证数据和测试数据
②模型数据：使用训练数据来构建使用相关特征的模型
③验证模型：使用你的验证数据接入你的模型
④测试模型：使用你的测试数据检查被验证的模型的表现
⑤使用模型：使用完全训练好的模型在新数据上做预测
⑥调优模型：使用更多数据、不同的特征或调整过的参数来提升算法的性能表现