特征工程

特征工程指的是最大程度上从原始数据中汲取特征和信息来使得模型和算法达到尽可能好的效果。

【特征工程具体内容包括：】

- 数据预处理
- 特征选择
- 特征变换与提取
- 特征组合
- 数据降维

一、特征工程常见的方法

1. 特征选择

数据预处理：一些前期的数据清洗和预处理工作，是对原始数据的基本整理和重塑。

在数据清洗、数据分析基本已完成。　　特征选择即选择与目标变量相关的自变量进行用于建模，也叫变量筛选

【特征选择基于两个基本面：】

- 特征是否发散，即该特征对于模型是否有解释力，如果特征是一成不变的（0方差），这样的特征是无用的。
- 特征是否与目标变量有一定的相关性。这一点要充分基于业务层面去考虑。

In [1]:
```
# 过滤法之方差筛选
from sklearn.feature_selection import VarianceThreshold
X = [[0, 0, 1], [0, 1, 0], [1, 0, 0], [0, 1, 1], [0, 1, 0], [0, 1, 1]]
sel = VarianceThreshold(threshold=(.8 * (1 - .8)))
sel.fit_transform(X)
```

Out[1]:
```
array([[0, 1],
       [1, 0],
       [0, 0],
       [1, 1],
       [1, 0],
       [1, 1]])
```

第一列值为0的比例超过了80%，在结果中VarianceThreshold剔除这一列

In [2]:
```
# 过滤法之卡方检验 通过卡方检验筛选2个最好的特征。
from sklearn.datasets import load_iris
from sklearn.feature_selection import SelectKBest
from sklearn.feature_selection import chi2
iris = load_iris()
X, y = iris.data, iris.target
X.shape
X_new = SelectKBest(chi2, k=2).fit_transform(X, y)
X_new.shape
```

Out[2]:
```
(150, 2)
```

In [3]:
```
# 嵌入法之基于惩罚项的特征选择法
from sklearn.svm import LinearSVC
from sklearn.datasets import load_iris
from sklearn.feature_selection import SelectFromModel
iris = load_iris()
X, y = iris.data, iris.target
print('原始数据特征维度：', X.shape)
lsvc = LinearSVC(C=0.01, penalty="l1", dual=False).fit(X, y)
model = SelectFromModel(lsvc, prefit=True)
X_new = model.transform(X)
print('l1惩罚处理之后的数据维度：', X_new.shape)
```

```
原始数据特征维度： (150, 4)
l1惩罚处理之后的数据维度： (150, 3)
D:\Python\python3.8\lib\site-packages\sklearn\svm\_base.py:1244: ConvergenceWarning: Liblinear failed to converge, incr
ease the number of iterations.
  warnings.warn(
```

In [4]:
```
# 嵌入法之基于树模型的特征选择法
from sklearn.ensemble import ExtraTreesClassifier
from sklearn.datasets import load_iris
```

```
from sklearn.feature_selection import SelectFromModel
iris = load_iris()
X, y = iris.data, iris.target
print('原始数据特征维度：', X.shape)
clf = ExtraTreesClassifier()
clf = clf.fit(X, y)
clf.feature_importances_
model = SelectFromModel(clf, prefit=True)
X_new = model.transform(X)
print('l1惩罚处理之后的数据维度：', X_new.shape)
```

```
原始数据特征维度： (150, 4)
l1惩罚处理之后的数据维度： (150, 2)
```

2.特征变换与特征提取

- 数据标准化：基于列 / 数据归一化：基于行
- 数据区间缩放
- 数值目标变量对数化处理（有必要的情况下）
- 定量特征二值化（有必要的情况下）
- 定性特征哑编码（one-hot）/大文本信息提取（效果类似于one-hot）

In [ ]:
```
# one-hot的两种方法
# sklearn onehotencoder
from sklearn.preprocessing import OneHotEncoder
from sklearn.datasets import load_iris
iris = load_iris()
OneHotEncoder().fit_transform(iris.target.reshape((-1,1))).toarray()
```

In [6]:
```
# pandas dummies 方法
import pandas as pd
pd.get_dummies(iris.target)
```

Out[6]:

	0	1	2
0	1	0	0
1	1	0	0
2	1	0	0
3	1	0	0
4	1	0	0
...	...	...	...
145	0	0	1
146	0	0	1
147	0	0	1
148	0	0	1
149	0	0	1

150 rows × 3 columns

3.特征组合

在单特征不能取得进一步效果的情况下可尝试不同特征之间的特征组合。
特别需要基于业务考量，而不是随意组合。

4.降维

适用于高维数据，成千上万的特征数量，但一般特征情况下不建议使用。

- PCA
- SVD
- LDA
- t-SNE

二.招聘数据的特征工程探索

In [7]:
```
import warnings
warnings.filterwarnings('ignore')
import numpy as np
import pandas as pd
```

In [ ]:
```
lagou_df = pd.read_csv('./lagou_data5.csv', encoding='utf-8',index_col=[0])
lagou_df.head()
```

In [9]:
```
# advantage和label这两个特征作用不大，可在最后剔除
# 分类变量one-hot处理
# pandas one-hot方法
pd.get_dummies(lagou_df['city']).head()
```

Out[9]:

	上海	其他	北京	南京	广州	成都	杭州	武汉	深圳
0	0	0	0	0	0	1	0	0	0
1	0	0	1	0	0	0	0	0	0
2	0	0	1	0	0	0	0	0	0
3	0	0	0	0	1	0	0	0	0
4	0	0	1	0	0	0	0	0	0

In [10]:
```
# 分类特征统一one-hot处理
cat_features = ['city', 'industry', 'education', 'position_name', 'size', 'stage', 'work_year']
for col in cat_features:
    temp = pd.get_dummies(lagou_df[col])
    lagou_df = pd.concat([lagou_df, temp],axis=1)
    lagou_df = lagou_df.drop([col], axis=1)

lagou_df.shape
```

Out[10]:
```
(1650, 54)
```

In [11]:
```
pd.options.display.max_columns = 999
lagou_df = lagou_df.drop(['advantage', 'label'], axis=1)
lagou_df.head(3)
```

Out[11]:

	position_detail	salary	上海	其他	北京	南京	广州	成都	杭州	武汉	深圳	O2O	企业服务	信息安全	其他	医疗健康	教育	数据服务	电子商务	硬件	移动互联网	金融	不限	博士	大专	本科	硕士	数据分析师	数据挖掘工程师	机器学习工程师
0	职位描述：工作职责：?1、负责新零售业务的数据分析工作，挖掘数据分析需求，制定并实施分析方案...	15000.0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0
1	职位描述：工作职责:方向一、经营分析/指标体系1. 参与公司核心策略的数据分析，基于策略逻辑...	32500.0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0
2	职位描述：职位描述：1、收集、处理用户海量数据，挖掘用户行为特征，为产品、运营提供参考依据；...	12500.0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	1	0	0

职位描述特征的信息提取

In [12]:
```
lagou_df2 = pd.read_csv('./lagou_data5.csv', encoding='utf-8', index_col=0)
lagou_df2 = lagou_df2[['position_detail', 'salary']]
lagou_df2.head(3)
```

Out[12]:

	position_detail	salary
0	职位描述：工作职责：?1、负责新零售业务的数据分析工作，挖掘数据分析需求，制定并实施分析方案...	15000.0
1	职位描述：工作职责:方向一、经营分析/指标体系1. 参与公司核心策略的数据分析，基于策略逻辑...	32500.0
2	职位描述：职位描述：1、收集、处理用户海量数据，挖掘用户行为特征，为产品、运营提供参考依据；...	12500.0

In [13]:
```
# 提取Python信息
for i, j in enumerate(lagou_df2['position_detail']):
    if 'python' in j:
        lagou_df2['position_detail'][i] = j.replace('python', 'Python')
```

In [14]:
```
lagou_df2['Python'] = pd.Series()
for i, j in enumerate(lagou_df2['position_detail']):
    if 'Python' in j:
        lagou_df2['Python'][i] = 1
    else:
        lagou_df2['Python'][i] = 0

lagou_df2['Python'].value_counts()
```

Out[14]:
```
1.0    1065
0.0     585
Name: Python, dtype: int64
```

In [15]:
```
lagou_df2['R'] = pd.Series()
for i, j in enumerate(lagou_df2['position_detail']):
    if 'R' in j:
        lagou_df2['R'][i] = 1
    else:
        lagou_df2['R'][i] = 0

lagou_df2['R'].value_counts()
```

Out[15]:
```
0.0    945
1.0    705
Name: R, dtype: int64
```

In [16]:
```
for i, j in enumerate(lagou_df2['position_detail']):
    if 'sql' in j:
        lagou_df2['position_detail'][i] = j.replace('sql', 'SQL')
```

```
lagou_df2['SQL'] = pd.Series()
for i, j in enumerate(lagou_df2['position_detail']):
    if 'SQL' in j:
        lagou_df2['SQL'][i] = 1
    else:
        lagou_df2['SQL'][i] = 0

lagou_df2['SQL'].value_counts()
```

Out[16]:
```
0.0    1203
1.0     447
Name: SQL, dtype: int64
```

In [17]:
```
lagou_df2['Excel'] = pd.Series()
for i, j in enumerate(lagou_df2['position_detail']):
    if 'Excel' in j:
        lagou_df2['Excel'][i] = 1
    else:
        lagou_df2['Excel'][i] = 0

lagou_df2['Excel'].value_counts()
```

Out[17]:
```
0.0    1551
1.0      99
Name: Excel, dtype: int64
```

In [18]:
```
lagou_df2['Java'] = pd.Series()
for i, j in enumerate(lagou_df2['position_detail']):
    if 'Java' in j:
        lagou_df2['Java'][i] = 1
    else:
        lagou_df2['Java'][i] = 0

lagou_df2['Java'].value_counts()
```

Out[18]:
```
0.0    1335
1.0     315
Name: Java, dtype: int64
```

In [19]:
```
for i, j in enumerate(lagou_df2['position_detail']):
    if 'linux' in j:
        lagou_df2['position_detail'][i] = j.replace('linux', 'Linux')
```

```
lagou_df2['Linux'] = pd.Series()
for i, j in enumerate(lagou_df2['position_detail']):
    if 'Linux' in j:
        lagou_df2['Linux'][i] = 1
    else:
        lagou_df2['Linux'][i] = 0

lagou_df2['Linux'].value_counts()
```

Out[19]:
```
0.0    1321
1.0     329
Name: Linux, dtype: int64
```

In [20]:
```
lagou_df2['C++'] = pd.Series()
for i, j in enumerate(lagou_df2['position_detail']):
    if 'C++' in j:
        lagou_df2['C++'][i] = 1
    else:
        lagou_df2['C++'][i] = 0

lagou_df2['C++'].value_counts()
```

Out[20]:
```
0.0    1165
1.0     485
Name: C++, dtype: int64
```

In [21]:
```
for i, j in enumerate(lagou_df2['position_detail']):
    if 'spark' in j:
        lagou_df2['position_detail'][i] = j.replace('spark', 'Spark')

lagou_df2['Spark'] = pd.Series()
for i, j in enumerate(lagou_df2['position_detail']):
    if 'Spark' in j:
        lagou_df2['Spark'][i] = 1
    else:
        lagou_df2['Spark'][i] = 0

lagou_df2['Spark'].value_counts()
```

Out[21]:
```
0.0    1237
1.0     413
Name: Spark, dtype: int64
```

In [22]:
```
for i, j in enumerate(lagou_df2['position_detail']):
    if 'tensorflow' in j:
        lagou_df2['position_detail'][i] = j.replace('tensorflow', 'Tensorflow')

    if 'TensorFlow' in j:
        lagou_df2['position_detail'][i] = j.replace('TensorFlow', 'Tensorflow')

lagou_df2['Tensorflow'] = pd.Series()
for i, j in enumerate(lagou_df2['position_detail']):
    if 'Tensorflow' in j:
        lagou_df2['Tensorflow'][i] = 1
    else:
        lagou_df2['Tensorflow'][i] = 0

lagou_df2['Tensorflow'].value_counts()
```

Out[22]:
```
0.0    1221
1.0     429
Name: Tensorflow, dtype: int64
```

In [23]:
```
lagou_df2 = lagou_df2.drop(['position_detail'], axis=1)
lagou_df2.head(3)
```

Out[23]:

	salary	Python	R	SQL	Excel	Java	Linux	C++	Spark	Tensorflow
0	15000.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0
1	32500.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
2	12500.0	1.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0

In [24]:
```
lagou_df = lagou_df.drop(['position_detail', 'salary'], axis=1)
lagou_df.head(3)
```

Out[24]:

	上海	其他	北京	南京	广州	成都	杭州	武汉	深圳	O2O	企业服务	信息安全	其他	医疗健康	教育	数据服务	电子商务	硬件	移动互联网	金融	不限	博士	大专	本科	硕士	数据分析师	数据挖掘工程师	机器学习工程师	深度学习工程师	15-50人	150-500人	2000人以上	50 15 人
0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	1	
1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0	0	0	0	1	
2	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	

In [25]:
```
lagou = pd.concat((lagou_df2, lagou_df), axis=1).reset_index(drop=True)
lagou.head(2)
```

Out[25]:

	salary	Python	R	SQL	Excel	Java	Linux	C++	Spark	Tensorflow	上海	其他	北京	南京	广州	成都	杭州	武汉	深圳	O2O	企业服务	信息安全	其他	医疗健康	教育	数据服务
0	15000.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	1.0	0.0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0
1	32500.0	1.0	1.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0

In [26]:
```
lagou.to_csv('lagou_featured.csv', encoding='utf-8')
```