Howework 3 Report

Student:鞠之浩 ID:P10942A08

Problem 1：One-by-one Feature Selection

(1) I use scikit-learn Univariate feature selection selectKbest function to pick 10 best features from gene, the score_function I used is f_classif and k gives 10. F_classif calculate the ANOVA(Analysis of Variance) F-value for the gene and label data.

(2) Result: the 10 features picked:

Hsa.692M76378 gene 1 "Human cysteine-rich protein (CRP) gene, exons 5 and 6.
"
Hsa.8147 M63391 gene 1 "Human desmin gene, complete cds.
"
Hsa.692M76378 gene 1 "Human cysteine-rich protein (CRP) gene, exons 5 and 6.
"
Hsa.36689 Z50753 gene 1 H.sapiens mRNA for GCAP-II/uroguanylin precursor.
Hsa.37937 R87126 3' UTR 2a 197371 "MYOSIN HEAVY CHAIN, NONMUSCLE (Gallus gallus)
"
Hsa.692M76378 gene 1 "Human cysteine-rich protein (CRP) gene, exons 5 and 6.
"
Hsa.1131 T92451 3' UTR 1 118219 "TROPOMYOSIN, FIBROBLAST AND EPITHELIAL MUSCLE-TYPE (HUMAN);.
"
Hsa.1832 J02854 gene 1 "MYOSIN REGULATORY LIGHT CHAIN 2, SMOOTH MUSCLE ISOFORM (HUMAN);contains element TAR1 repetitive element ;.
"
Hsa.6814 H08393 3' UTR 2a 45395 COLLAGEN ALPHA 2(XI) CHAIN (Homo sapiens)
Hsa.2456 U25138 gene 1 "Human MaxiK potassium channel beta subunit mRNA, complete cds.
"

Code:

```
# define feature selection
fs = SelectKBest(score_func=f_classif, k=10)


# In[11]:


KBest = fs.fit_transform(gene,label)


# In[13]:


#找index position
position = []
for i in range(len(KBest[0])):
    #print(np.where(gene[0] == test_selected[0][i])[0][0])
    position.append(np.where(gene[0] == KBest[0][i])[0][0])
position = np.array(position)
position
```

Problem 2：Subset-Based Feature Selection

1.(a)The metaheuristic I choose is GA. I use the GeneticSelectionCV model from genetic_selection package.

(b)My objective function is Decision Tree Classifier.

(c)These are the tunable parameters I used.

```
model = GeneticSelectionCV(estimator, #A supervised learning estimator with a `fit` method.
                           cv = 5, #Determines the cross-validation splitting strategy.
                           verbose = 0, #Controls verbosity of output.
                           scoring = "accuracy", #a scorer callable object / function with signature
                           max_features = 10, #The maximum number of features selected.
                           n_population = 100, #Number of population for the genetic algorithm.
                           crossover_proba = 0.5, #Probability of crossover for the genetic algorithm.
                           mutation_proba = 0.2, #Probability of mutation for the genetic algorithm.
                           n_generations = 40, #Number of generations for the genetic algorithm.
                           crossover_independent_proba = 0.1, #Independent probability for each attribute to be exchang
                           mutation_independent_proba = 0.05, #Independent probability for each attribute to be mutatec
                           tournament_size = 3, #Tournament size for the genetic algorithm.
                           n_gen_no_change = 10, #terminate optimization when best individual is not changing in all of
                           caching = True, #If True, scores of the genetic algorithm are cached.
                           n_jobs = -1 #Number of cores to run in parallel.
                          )
```

2.Result: the 7 features picked:

Hsa.1190	T74556	3' UTR	1	84680 "ATP SYNTHASE ALPHA CHAIN, MITOCHONDRIAL PRECURSOR (HUMAN);. "
Hsa.9817	T50334	3' UTR	2a	72201 14-3-3-LIKE PROTEIN GF14 OMEGA (Arabidopsis thaliana)
Hsa.823	M16937	gene	1	"Human homeo box c1 protein, mRNA, complete cds. "
Hsa.1588	U09587	gene	1	"Human glycyl-tRNA synthetase mRNA, complete cds. "
Hsa.1620	U18422	gene	1	"Human DP2 (Humdp2) mRNA, complete cds. "
Hsa.743	T62884	3' UTR	1	86044 6-PHOSPHOFRUCTO-2-KINASE (HUMAN);.
Hsa.1209	T41204	3' UTR	1	62344 "P14780 92 KD TYPE V COLLAGENASE PRECURSOR ,. "

Code:

Problem 3：Arima Forecast

(1) The Arima parameters (p, d, q, P, D, Q, s) are (0, 1, 0, 2, 1, 0, 12) respectively.

The MSE is 229359.6859874526

(2)