

任务一

- 首先将所有的AMT_CREDIT都转化为整型
- 新建一列“CREDIT_RANGE”，用于存放AMT_CREDIT所属区间，形式为 $((AMT_CREDIT // 10000) * interval, (AMT_CREDIT // 10000 + 1) * 10000)$
- 根据CREDIT_RANGE对数据进行分组，并统计每一组的个数
- 返回结果

2. 编写Spark程序，统计application_data.csv中客户贷款金额AMT_CREDIT比客户收入AMT_INCOME_TOTAL差值最高和最低的各十条记录。

- 新建一列DIFFERENCE，用于存放AMT_CREDIT和AMT_INCOME_TOTAL的差值
- 根据DIFFERENCE进行降序排列，取前十个记录，这是差值最高的十条记录
- 根据DIFFERENCE进行升序排列，取前十个记录，这是差值最低的十条记录
- 将结果合并为一个数据框，并选择要保留的列进行输出

[illegible]

任务二

1. 基于Spark SQL，统计所有男性客户（CODE_GENDER=M）的小孩个数（CNT_CHILDREN）类型占比情况

- 先将原始数据中CODE_GENDER=M的部分筛选出来，记作gender_filtered
- 然后计算gender_filtered中CNT_CHILDREN的分布

```
children_count = gender_filtered.groupBy("CNT_CHILDREN").count()
```

- 根据gender_filtered可以得出所有男性客户的总人数
- 在children_count新建一列ratio，值为count/总人数，结果保留4位小数
- 根据CNT_CHILDREN进行升序排列，输出结果

```
1 0,0.6693
2 1,0.2157
3 2,0.0991
4 3,0.0138
5 4,0.0016
6 5,0.0003
7 6,0.0001
8 7,0.0
9 8,0.0
10 9,0.0
11 11,0.0
12 14,0.0
```

2. 基于Spark SQL，统计每个客户出生以来每天的平均收入（avg_income）=总收入（AMT_INCOME_TOTAL）/ 出生天数（DAYS_BIRTH），统计每日收入大于1的客户，并按照从大到小排序

- 在原始数据框新增一列avg_income，表示每个客户出生以来每天的平均收入
- 筛选出avg_income>1的部分，并根据avg_income降序排列
- 选择需要保存的列，输出

	A	B	C	D	E	F	G	H	I	J
1	SK_ID_CURR	avg_income								
2	114967	9274.67300832343								
3	336147	1146.21051961284								
4	385674	996.236440115121								
5	190160	547.945205479452								
6	219563	417.517164594544								
7	310601	373.634080590238								
8	157471	360.432519022827								
9	252084	348.999534667287								
10	199821	269.654841802493								
11	337151	243.757109582363								
12	141198	243.623676612127								
13	429258	241.659394508962								
14	196091	240.761877585961								
15	317748	240.448837830617								
16	432980	239.565587734242								
17	217276	235.320484087853								
18	445335	234.308435103664								
19	387126	230.465320456541								
20	304300	223.583968201391								
21	123587	207.249674902471								
22	399467	195.921921486104								
23	441639	192.455735180908								
24	440768	192.040968739998								
25	225210	188.75388133737								
26	206341	186.00165334803								
27	134526	183.688464364438								
28	214063	180.082716551444								
29	336135	177.464788732394								
30	111903	174.135128859995								
31	269498	172.602739726027								
32	194130	172.005198379329								
33	431111	166.759310728182								

任务三 基于Spark MLlib 或者Spark ML编写程序对贷款是否违约进行分类，并评估 实验结果的准确率。

1. 数据集分割

- 首先将原始数据清洗，将带有缺失值的行全部删除
- 利用sklearn库中的train_test_split方法对清洗后的数据集进行随机分割，训练集与测试集的大小之比为4：1

2. 特征工程

- 对文本和离散特征进行 StringIndexer 和 OneHotEncoder 处理
- 过滤掉具有大量唯一值的列，因为这些特征大概率不会给模型带来泛化能力上的提升
- 将所有处理过的特征组合成一个向量，便于后续模型训练
- 特征标准化

ps:模型实例化之后，即可与上述步骤组合成Pipeline,方便模型训练与预测

3. 不同模型的选择与性能表现

- 直接从pyspark.ml.classification 中选择想要的模型

逻辑回归

```
Accuracy: 0.9194510836869746  
F1 Score: 0.8809446106263878
```

随机森林

```
Accuracy: 0.9195323805342829  
F1 Score: 0.8809851893362669
```

支持向量机

```
Accuracy: 0.9194510836869746  
F1 Score: 0.8809446106263878
```

可以看到这三个分类模型在该数据集上的性能都是差不多的。