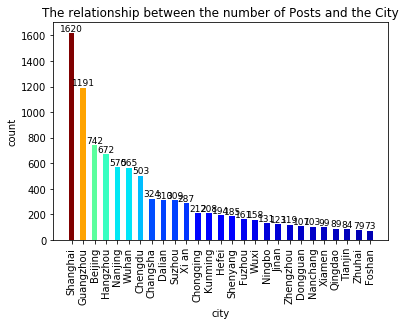
Data analysis

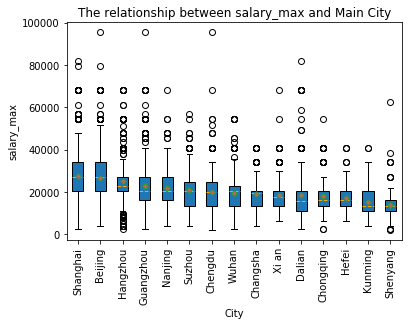
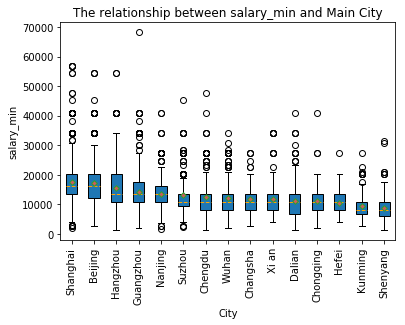
2.1

2.1.1 visualization

本章节使用python语言在jupyter notebook中绘制图片来探寻与岗位数量相关的因素以及与薪资相关联的因素。通过绘制条形图与箱型图来查找其中的关系。这里使用到了python中的3个工具包，它们分别是matplotlib，pandas和numpy。

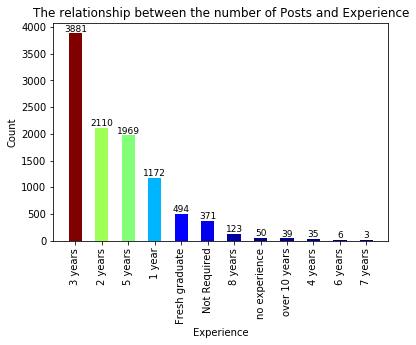


由于空间的限制，本图仅显示前25个城市的java岗位数量。通过条形图可以直接发现岗位的需求量与城市的规模有着密切的相关性。直辖市的需求量大于省会城市的需求量大于一般地级市的需求量。上海的java岗位需求量远远领先于中国的其他城市。需求量越大说明该城市对于掌握java技能的人来说机遇也就越大，同时也更有利于一个程序员的未来发展。虽然抓取到的数据仅仅只是一个多月的数据记录但也足以反映出当地企业的需求量。

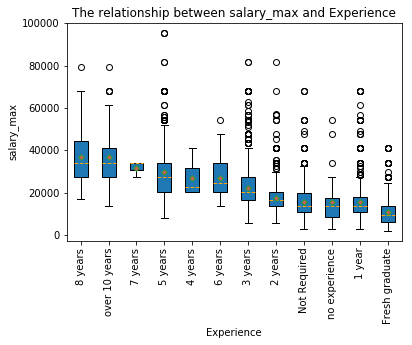
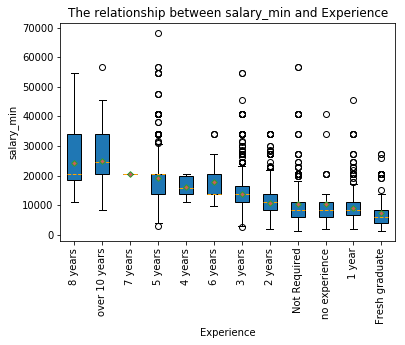


通过上述的岗位数量与城市之间的关系，这里再来看看薪资与城市之间的关系。同样由于空间的原因这里仅仅分析前15个城市的情况。为了更好更全面的观察，这里运用箱型图来展示。大致情况与之前的岗位数量分布相类似。然而广州从第二位下降到了第四位。因此虽然这座城市的需求比较大但它的整体薪资并没有那么高。除此之外它的岗位需求数甚至是后面的1.5倍或更高。杭州的薪资水平是仅次于前面两个直辖市的，而且它的薪资更加地分散。每一座城市都存在异常值，这些应该指的是那些高端技术职业。虽然杭州的超高薪并没有北京及广州那么高，但它的数量却是最多的。因此可以说杭州更加欢迎高端技术型人才扎根于此。

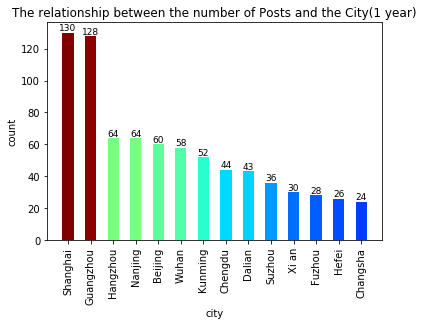
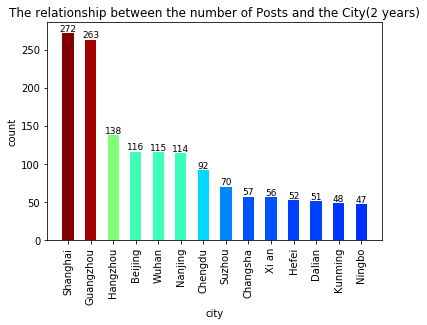
接下来来看经验、学历与城市之间的关系。

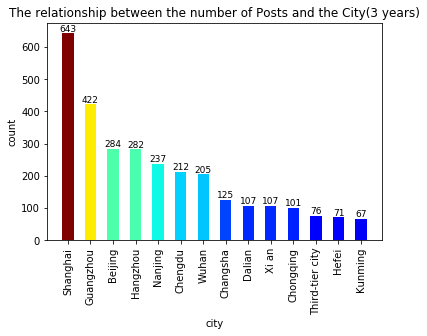
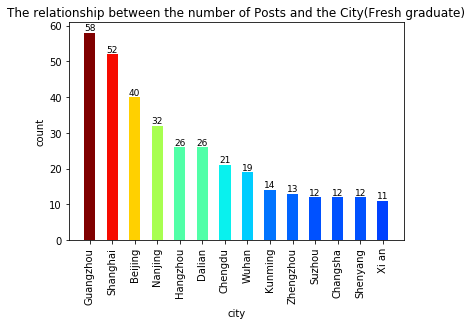


从整体上来看最受欢迎的是具有3年经验项目经验的程序员，它的人数几乎是拥有两年经验和5年经验的人的两倍。因此可以说明拥有三年以上项目经历的更受公司的青睐。对于应届生或者说是缺乏经验的人来说公司提供的岗位数量不能算多也不能算少，但也可以说明51job这个网站对于这一类的人群来说是一个可以进行寻找工作的工具。同时也对缺乏工作经验的年轻人来说在目前这个大的就业环境之中工作经验是硬实力，不要去随随便便地更换一份工作除非你已经很有实力与别人进行竞争。6年以上经验的需求量则相对较少，这也能反映出人们在一个相对稳定的工作环境中不太愿意去更换一个新的公司。



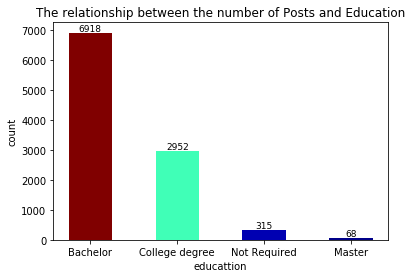
这里使用箱型图来更清晰地反映薪资与工作经验之间的关系。通过最低工资与最高工资的箱型图我们就可以发现随着工作经验的增长程序员所获得的工资也在增长。因此工作经验与薪资是正相关的。对于一所公司来说肯定是希望员工的工作经验越高越好，但为了得到这一类的员工它所要付出的代价当然也是相当大的。对于一个刚入业的新人来说，良好学习的机会和待遇一样重要。同样的各阶段也存在很多异常值。

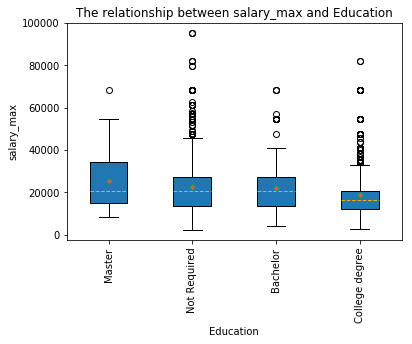
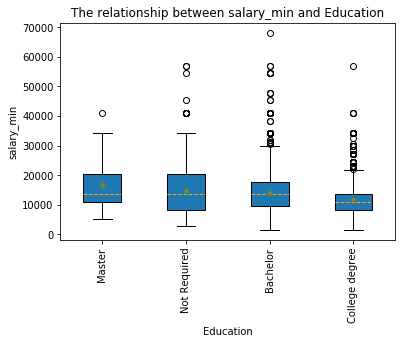
 

这四个子图分别是不同的城市对于不同工作经验的人的需求量。通过图片可以发现对于应届生与工作经验仅为1年的年轻人来说，广州上海的需求量远超其他城市。这里也就解释了为什么广州的岗位需求量很高但它的总体薪资却很低这一现象了。因为我们之前也得出了一个结论就是薪资与工作经验的正相关性。对于各个阶段来说伤害依然是广大程序员的首选之地不仅仅因为它的总体薪资相对较高还因为它欢迎各个阶段的人去工作。但我们也知道大城市的消费水平高从而导致工作压力大这一事实，所以对于初入职场的年轻人杭州南京武汉等省会城市也是不错的选择。

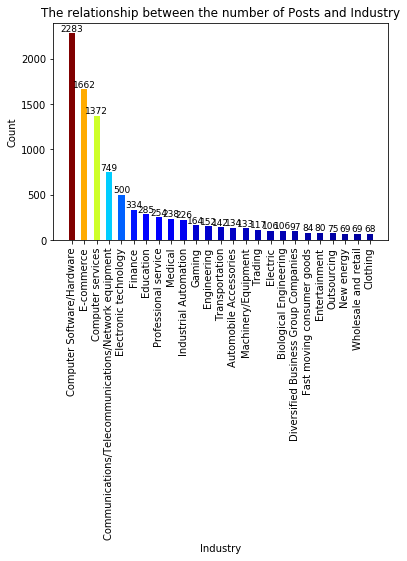
以下再来看看学历要求需求量。



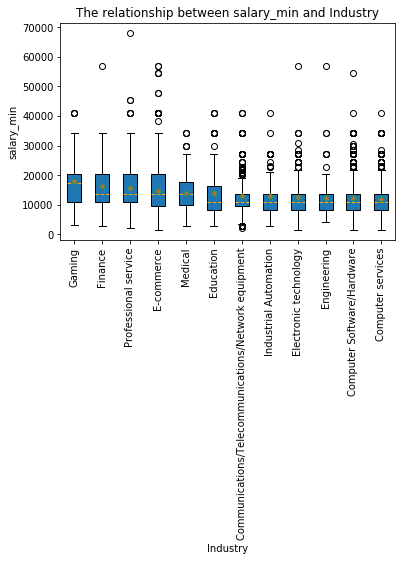
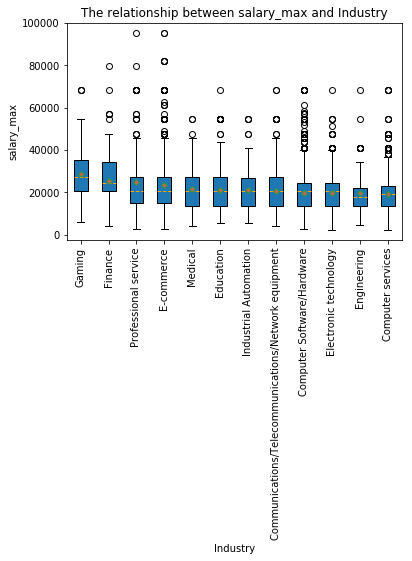
通过条形图就可以发现本科学历的需求量是最多的而且数量上是大专学历的两倍还多。仅有2.99%的岗位是对应聘者没有学历要求的。而学历为硕士及以上的岗位仅仅只有68个可以看出对于java岗位来说工作经验比学历更加重要。



通过箱型图来找寻学历与薪资之间的关系，通过上面两个图片所反映出来的情况来看，学历与薪资是正相关的。换句话说，一个程序员的学历越高他所获得工资也就越高。此外还可以看出无学历要求的岗位与学历为本科的岗位的薪资向接近因此也可以将这一类的学历划归到学历为本科的人群之中。

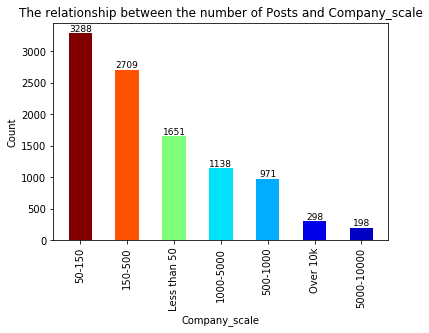


通过条形图可以看到不同的产业对于java岗位的需求是不一样的，其中与计算机相关的行业占据了主要地位，其次是电子技术、金融、专业服务等行业。剩下的产业对于java程序员的需求量都很小。

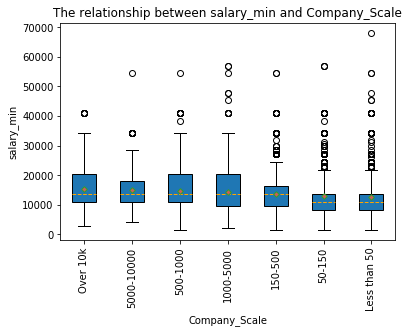
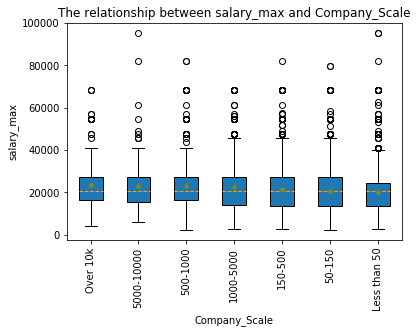


这里使用箱型图来观察不同的产业之间的薪资结构。犹豫空间的因素这里仅显示数量前12的行业情况。与之前分布情况不一样，这里计算机相关的行业薪资却是排名靠后。排名靠前的是游戏行业、金融行业和专业服务行业。这些行业对于java程序员的需求量不高但给出的工资却相对较高。这里反映出了一个现象就是单一型人才获得的薪资较低，而复合型程序员在如今的职业环境中更加具有竞争力。

最后来看看有关公司规模的部分



通过条形图可以发现公司人数小于150人和公司人数大于15人的数量是相近的。大部分的公司都是中小规模的。



通过箱型图可以明显看到公司的规模越大平均工资也就越大。倒数后三个公司规模都是小于150人的企业。因此这也符合了大部分人的心态，大家都希望能够进一个大公司开始自己的职业。

岗位要求的关键字提取

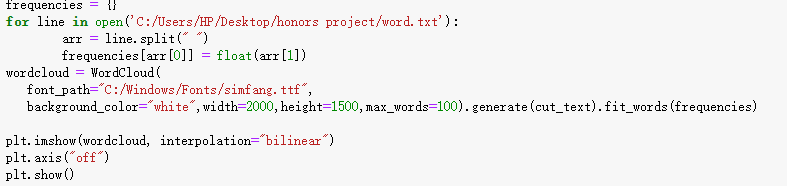
通过使用python中的jieba包进行关键词的提取并使用wordcloud包根据词频来生成词云。首先对之前获得的job\_demand进行整合。这里需要将每一条数据合并成一个字符串。通过调用jiebao包来实现自动划分高频词。并且根据每个词出现的次数统计它们的频率。这里由于空间的原因仅显示前23条数据。



通过统计的数据就不难发现出现最多的是java这个词，但这里这个词的出现并没有实际的意义。Mysql,Oracle和Sql是出现次数最多的几个词，它们都是和数据库相关的。这表明了成为一个java程序员还需要熟练使用数据库的能力。Spring，Mybatis，springmvc和spring boot这四个都是出现次数最多的框架名词。这说明它们都是目前为止非常主流的java 框架。通过以上的结论为的就是展示目前各大企业具体需要的是什么，也给正在学习java技术的大学生一个启示避免学习了偏冷门的知识而没有跟上时代的潮流。



通过先前生成的词频在jupyter notebook中使用python代码生成词云。

这里可以看出字体的大小与出现的频率是正相关的。因此，出现次数越多生成的字体也就越醒目。

2.2 建立模型

使用之前处理好的数据建立来进行数据分析。这里将用到三种不同的算法来进行，分别是分类算法，关联算法和聚类算法并从每一个算法中选择一定数量的规则。

2.2.1 分类算法

分类算法是一个有监督的机器学习算法，确定目标数据中哪些类别是已知的，分类的过程就是把每一条记录归到相对应的类别中去。我将使用两种方法来实现分类算法。第一种方法是在python中使用scikit-learn来建立决策树模型。第二种方法是在weka中运用J48算法来进行实现分类算法。J48的本质也是决策树模型，但它在ID3的基础上进行了更多的剪枝。我期望能够通过这一算法寻找获得低收入和中等收入的一种组合。

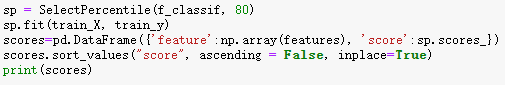
2.2.1.1 在python中实现决策树

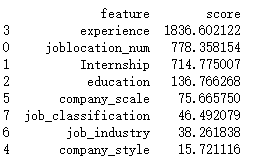
首先，在python中添加一个新的工具包sklearn。这里使用到的数据都是之前整合之后的数值型数据。虽然没整合的数据也可以使用，只需要在分析之前进行独热编码。但为了简化过程所以还是使用已经整合好的数据。

第一步进行数据读取，并对读取完的数据划分训练集与测试集。这里选择75%的数据作为训练数据集，25%的数据作为测试数据集。



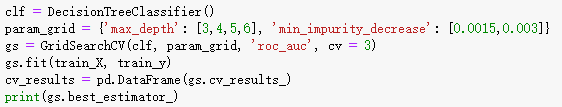
第二步，定义变量名称相对应的索引并且创建特征选择器，计算每个预测变量对于目标变量的F值。



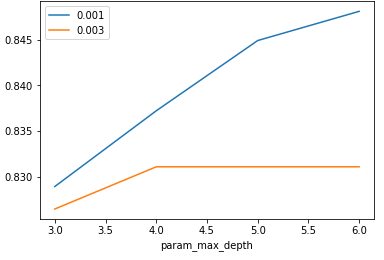


结果表明公司性质与预测的结果关系不大。换句话说它们不具备相应的预测能力。因此选择前7个重要性最高的特征变量。

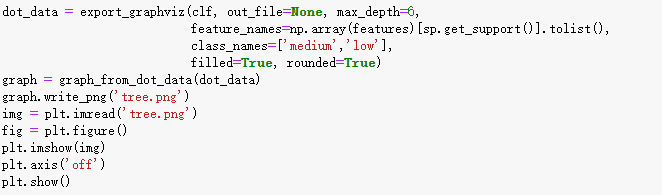
第三步，进行参数调优。设置网格搜索超参数最大深度为3,4,5,6和最小掺杂度减少比例的区间范围是0.001-0.003，设置重采样策略为3折交叉验证。

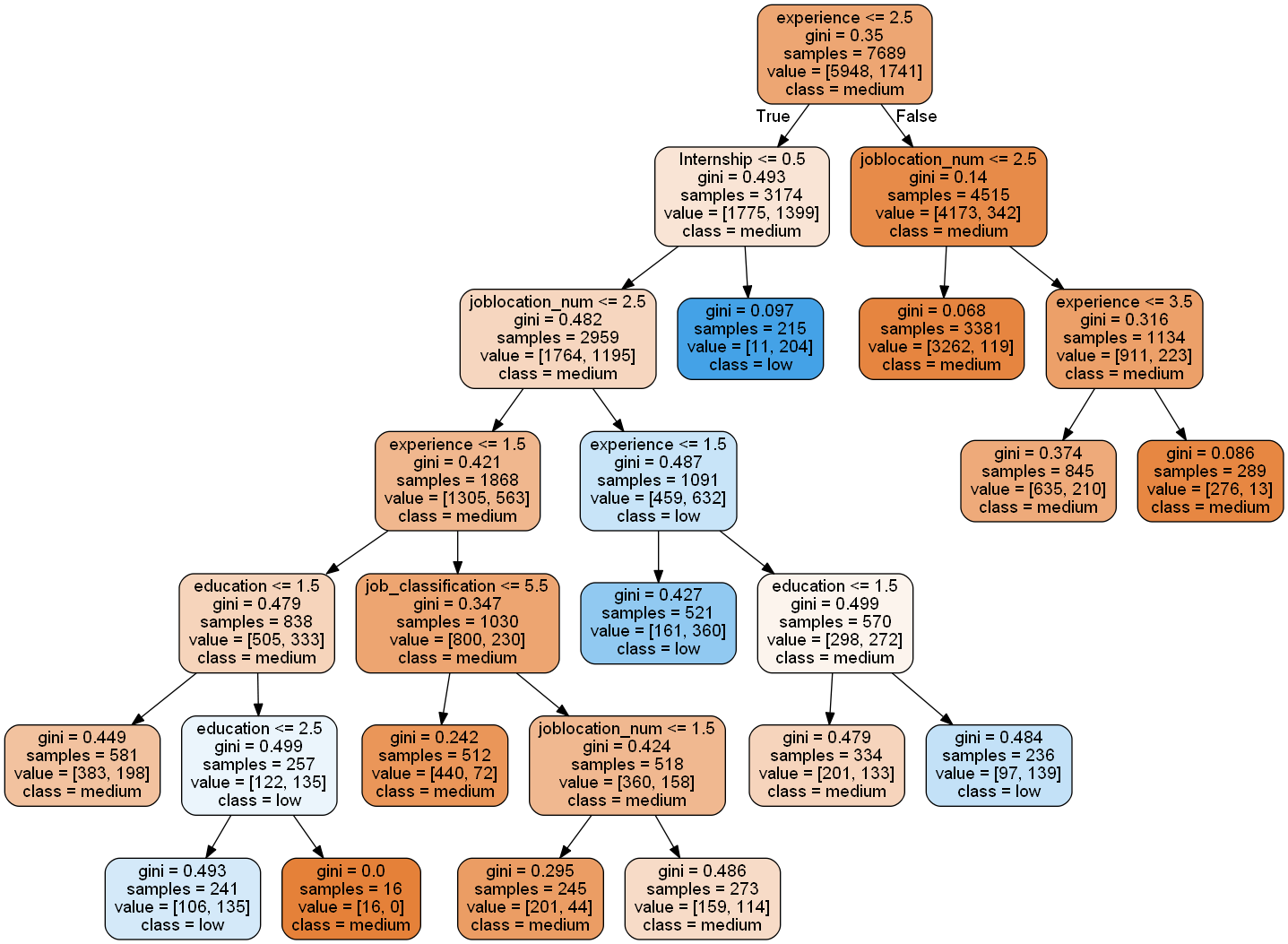


结果表明在最大深度为6的时候可以获得最优的预测效果。



最后使用搜索深度参数6生成决策树模型。





依据这个决策树我找到三条规则：

Rule1: IF experience<=2.5 AND Internship >0.5 Then class=low (204/215)

该规则表明工作经验小于两年并且是个实习生那么他所获得的薪水值就会很低，215条记录中有204条被正确标记。这点也很符合正常的逻辑，一个缺乏经验的实习生的收入必然是不会很高的。

Rule2: IF experience>2.5 AND joblation\_num<=2.5 Then class= medium(3262/3381)

改规则表明工作经验大于等于三年并且是在直辖市或是省会城市那么他所获得的薪水就是中高水平的。总共有3381条记录其中有3262条被正确标记。这也是所有的规则中数量最多的一个，也可以说明这个规则被广发的运用。

Rule3: IF joblocation\_num>2.5 AND experience>3.5 Then class=medium(276/289)

该规则说明了一个java岗位即使是在二三线城市，但是候选人的工作经验大于等于4年那么他也会获得中高收入。该规则有289条其中276条被正确标记。该规则表明经验的比工作城市更加重要。

Rule4: IF Internship<=0.5 AND joblocation<=2.5 AND experience<=1.5 AND

job\_classification <=5.5 Then calss=medium(440/512)

该规则表明在一线城市的非实习岗位即使候选人经验不足，但他从事的是软件工程师或者高级软件工程师或者java开发人员，那么他的收入也不会低。该规则有512条记录，其中440条被正确标记。改规则表明了一个好的岗位也是极其重要的选择。

最后，使用测试数据集来检验模型的可靠度p=84.6%。这结果表明该模型的可信度是相当高的。

2.2.1.2 weka实现J48(C4.5)

Weka的操作过程相对于python来说更加方便。因为它被高度封装在了程序内，仅仅需要导入已经整合好的数据就可以实现。这里首先将CSV格式的文件经过weka修改为arff后缀的文件。这么做的目的是为了方便文件的读取。该算法的本质就是ID3算法。在test option中同样选择use training set。然后更改默认参数，因为我发现默认的参数树叶过多。这里将置信度改为0.1，其余参数则保持不变。得到树叶为70，树的大小为85，数的深度为5。该模型的准确度为83.87%。这里同样找到了四条置信度及数量较高的规则：

Rule1: IF joblocation\_num=First-tier city AND Other First-tier city Then class=medium(1362/1655)

规则表明在一线城市的收入大都是中高水平，这里有1655条数据其中1362条数据被正确标记。

Rule2: IF joblocation\_=Second-tier city AND education=College degree Then class=low(176/253)

该规则表明在二线城市并且学历为大专，那么他的薪资水平为低，这里有253条数据被记录其中有176条数据被正确标记。

Rule3: IF experience=3 years AND 5 years Then class=medium(5850/6304)

该规则表明工作经验在3至5年的候选人那么他必将获得中高水平的薪资。这里有6304条数据其中5850条被正确标记。该规则的使用范围也是最广泛的一个。

Rule4: IF experience=1 year AND joblocation\_num=Second-tier city and Third-tier city Then class=low(478/625)

该规则表明缺乏工作经验并且在二三线城市那么他的薪资是低的。该规则有625条数据其中478条数据被正确标记。

2.2.2 关联规则

在关联算法中，我选择用Apriori算法来实现数据挖掘。Apriori算法也是关联算法中最经典的算法。Apriori is based on breadth-first search. 在该算法中我使用的数据是我在之前小节中所描述过的名值数据集。在使用默认参数进行关联之后，我发现由于过高的confidence，在结果中显示出的规则数量并没有达到预设的10个。因此，我对默认的参数值进行了调整。我将minMetric调整为了0.85,并且将numRules调整为了15，而其他的参数都保持默认值。

以下规则是我在15条规则中选取的5条。

education=Bachelor job\_classification=Senior software engineer 2283 ==> Salary\_grade=medium 2106 conf:(0.92)

该规则表明了学历为本科的高级软件工程师的薪资水平在中高水平，2283条数据中有2106条被正确标记，而且置信度达到了0.92，具有高度可用性。

education=Bachelor experience=3 years 2657 ==> Salary\_grade=medium 2428 conf:(0.91)

该规则表明了拥有三年经验的本科生他的薪资为中高水平，2428条被正确标记。该规则的数量与可信度都相当高。

joblocation\_num=First-tier city education=Bachelor 2560 ==> Salary\_grade=medium 2364 conf:(0.92)

改规则表明一线城市的本科生java程序员的收入也在中高水平。2364条被正确标记。该规则的数量与可信度都相当高。

joblocation\_num=First-tier city experience=3 years 1349 ==> Salary\_grade=medium 1328 conf:(0.98)

改规则表明一线城市的三年经验的java程序员的收入也在中高水平。该规则的数量与可信度都相当高。

joblocation\_num=First-tier city job\_classification=Senior software engineer 1214 ==> Salary\_grade=medium 1175 conf:(0.97)

改规则表明一线城市的高级软件工程师的薪资水平为中高水平。该规则的数量与可信度都相当高。

2.2.3 聚类算法

聚类算法是研究分类问题的一种统计分析方法。在weka中我选择使用simplekmeans算法来进行聚类分析。它是属于无监督学习。它的算法思想是先任意取k个值作为中心点，然后比较数据集中每个点到各个中心点的欧几里得距离。最后，把距离近的划分为一个聚类。在默认聚类数量为2的时候就已经成功获得薪资水平为low和 的部分，但精确度较差。因而我将聚类数量调整为了4。每一个cluster就代表了一个规则。



在cluster0中就表明了这一类人是可能获得低等工资的。在非北上广的一线城市拥有两年工作经验的java软件工程师并且在中小规模工作，那么他有可能获得较低的工资。这类人总数达到2447十分接近原始数据中的2308.。

在后三个cluster中都表明了这三类人有极大可能是中高收入的。其中1和2的情况非常相似。他们都是在一线城市工作的软件工程师，分别在中规模的计算机服务公司和小规模的计算机软硬件公司工作。

第三类人有极大可能获得中高收入。这类人是在二线城市小型计算机软硬件公司工作的高级软件工程师，虽然他们的学历也仅为大专学历。