Лабораторная работа 2
по дисциплине «Data Mining»

В R деревья решений реализуются с помощью двух основных пакетов: *party* и *rpart.* «Случайный лес» реализуется пакетом *randomForest.*

Задание 1.
Имеется фрагмент базы данных об анкетировании клиентов турфирмы.

Разделите данные на обучающую и тестовую выборки по 70% и 30%, либо используя другие проценты по своему усмотрению.

```
> library(party)
> set.seed(1234)
> ind <- sample(2, nrow(myData), replace=TRUE, prob=c(0.7, 0.3))
> trainData <- myData[ind==1,]
> testData<- myData[ind==2,]
```

Постройте дерево решений на основе пакета party, выбрав интересующую Вас зависимую переменную. Поясните Ваш выбор. В частности, какую неизвестную информацию об исходных данных позволит получить построение дерева решений для этой переменной? Обозначьте зависимую переменную как Y, и классификационные переменные как X1, X2, и т.д., либо другим удобным способом.

```
> library(party)
> myFormula<- Y~X1+X2+X3+...+Xk
> myData_ctree<-ctree(myFormula, data=trainData)
```

Приведите матрицу классификации.

```
> table(predict(myData_ctree),
trainData$имя_выбранной_результативной_переменной)
```

Приведите результаты построения дерева решений в виде таблицы.

```
> print(myData_ctree)
```

Опишите последовательность проведения классификации с точки зрения переменных.

Приведите дерево решений на графике в общем виде.

```
> plot(myData_ctree)
```

Приведите дерево решений на графике в упрощенном виде.

```
> plot(myData_ctree, type="simple")
```

Дайте интерпретацию дерева решений с точки зрения листьев (терминальных узлов). Опишите вероятности для каждого класса результативного признака, указав количество наблюдений в каждом классе, с точки зрения значений параметров классифицирующих признаков.

Сведите полученные выводы в небольшую аналитическую записку о результатах классификации.

Примените полученное дерево решений для тестовых данных. Дайте общую характеристику ошибкам классификации.

```
> testPred<-predict(myData_ctree, newdata=testData)
> table(testPred, testData$имя_выбранной_результативной_переменной)
```

Задание 2.

Постройте дерево решений с помощью пакета rpart, выбрав другую зависимую переменную. Поясните выбор зависимой переменной с точки зрения получения какой-либо новой информации об имеющихся данных.

Измените обозначения факторных и результативных переменных в исходных данных, соответственно.

Разделите данные на обучающую и тестовую выборку по 80% и 20%, соответственно.

```
> set.seed(1234)
> ind<-sample(2, nrow(myData), replace=TRUE, prob=c(0.8, 0.2))
> tourism.train<-myData[ind==1,]
> tourism.test<-myData[ind==2,]

> library(rpart)
> myFormula<-Y+X1+X2+X3+...+Xk
> tourism_rpart<-rpart(myFormula, data=tourism.train,
control=rpart.control(minsplit=10))
```

Приведите результаты построения дерева решений в виде таблицы и графика с подписями.

```
> print(tourism_rpart)
> plot(tourism_rpart)
> text(tourism_rpart, use.n=T)
```

Приведите таблицу с построенными деревьями, в которой указаны ошибки классификации.

```
> print(tourism_rpart$cptable)
```

Приведите дерево с наименьшим количеством ошибок классификации в виде таблицы и графика с подписями.

```
> opt<-which.min(tourism_rpart$cptable[, "xerror"]
> cp<-tourism_rpart$cptable[opt, "CP"]
> tourism_prune<-prune(tourism_rpart, cp=cp)
> print(tourism_prune)

> plot(tourism_prune)
> text(tourism_prune, use.n=T)
```

Приведите точечный график результатов классификации. Сделайте выводы о подгонке модели классификации с точки зрения расположения точек на графике.

```
> Ytourism_pred<-predict(tourism_prune, newdata=tourism.test)
> print(Ytourism_pred)
> xlim<-range(myData$имя_результативного_признака)
> plot(Ytourism_pred~Y, data=tourism.test, xlab="Факт", ylab="Классификация",
ylim=xlim, xlim=xlim)
> abline(a=0, b=1)
```

Задание 3.

Имеются данные о заявлениях на выдачу кредита.

Какие из этих переменных количественные? Какие качественные?

Какие из исходных переменных дискретные? Какие непрерывные?

Какие из исходных переменных номинальные? Какие порядковые?

Разделите данные на обучающую выборку (Train) и на данные, которые необходимо верифицировать (Verify), предварительно рассортировав таблицу.

```
> attach(myData)
> forestData<-myData[order(NNSET),]
> detach(myData)
```

необходимо убрать переменную NNSET (которая содержит коды "train" и "verify")

```
> forestData$NNSET<-NULL
```

Разделим данные на две категории.

```
> trainData<-forestData[1:30,]
> testData<-forestData[31:60,]
```

Постройте модель классификации по алгоритму «Случайный лес», сформировав 100 деревьев.

```
> library(randomForest)
> rf<-randomForest(GROUP~ . , data=trainData, ntree=100, proximity=TRUE)
```

Приведите график построения модели.

```
> plot(rf)
```

Достаточно ли этого количества деревьев для классификации? Как вы это определили?

На основе скольких деревьев была получена устойчивая модель классификации?

Приведите матрицу классификации для обучающей выборки.

```
> print(rf)
```

Каков процент ошибок классификации для отклоненных заявок?

Каков процент ошибок классификации для одобренных заявок?

Приведите классификационные переменные в порядке убывания информативности (значимости) в виде списка и графика.

```
> importance(rf)
```

```
> varImpPlot(rf)
```

Проведите проверку анкет для верификации (тестовая выборка).

```
> mydataPred<-predict(rf, newdata=testData)
```

Приведите матрицу классификации для тестовой выборки.

```
> table(mydataPred, testData$GROUP)
```

Приведите график предельных ошибок классификации (Predictions margin) и интерпретируйте ошибки классификации с точки зрения разницы между количеством голосов за верную классификацию против максимального количества голосов за другие варианты.

```
> plot(margin(rf, testData$GROUP))
```

Сохраните построенную модель.

```
> saveRDS(rf, "forest.rds")
```

Напишите краткую аналитическую записку, которая содержит общие выводы о качестве построенной модели и результатах классификации.

Загрузите построенную модель «Случайного леса».

```
> my_model<-readRDS("forest.rds")
```

Примените загруженную модель для поступивших новых заявок на выдачу кредита (назовите таблицу в R, например, newPred)

```
> predict(my_model, newdata=newPred)
```

Либо можно сразу записать эти результаты в новую таблицу:

```
> newPred$GROUP<-predict(my_model, newdata=newPred)
```

Приведите результаты классификации новых анкет.