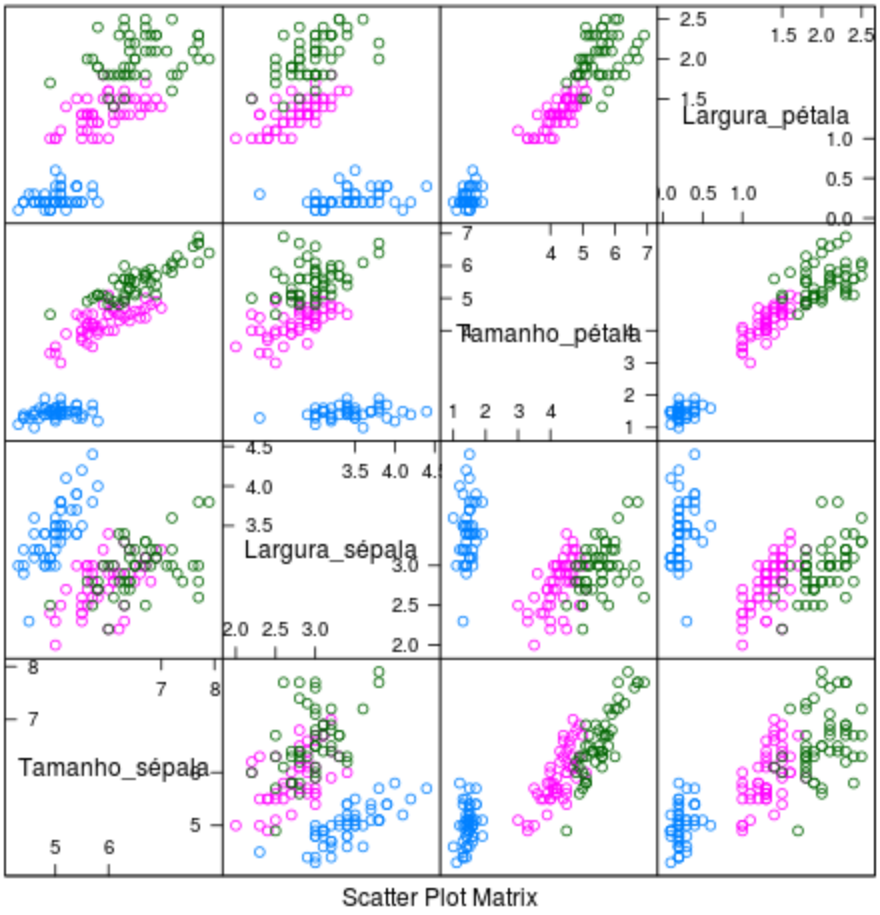
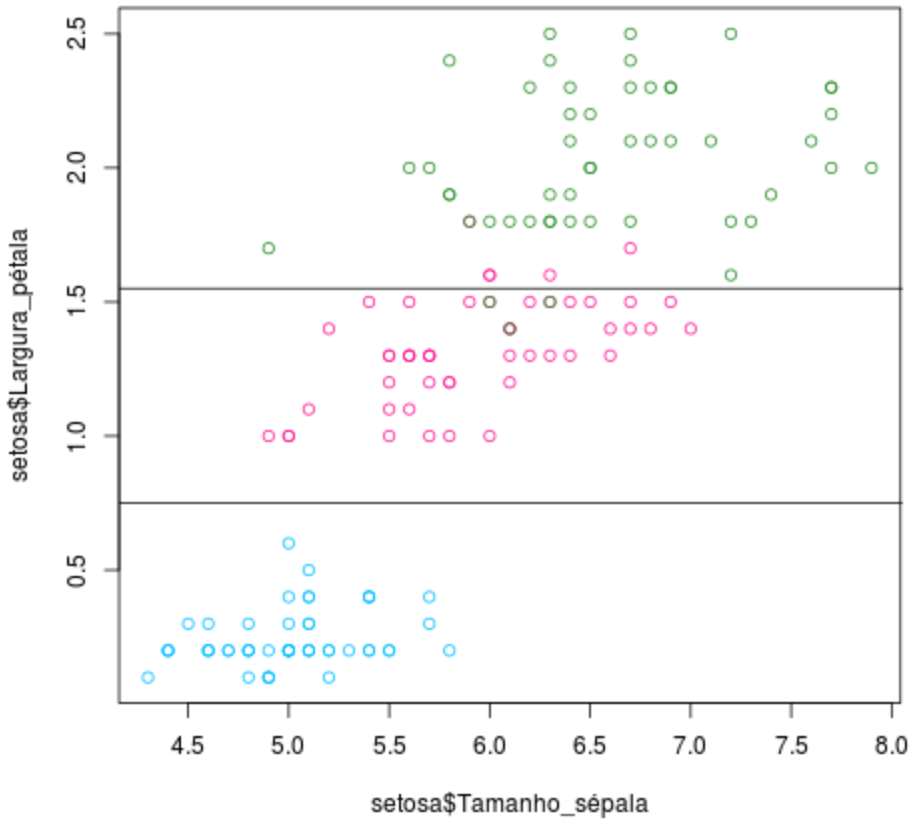
Gabarito!

Q1:



Scatter Plot Matrix

Q2.a: exemplo



Código R

```
dados <- read.table("plantas-iris.txt", header = TRUE, sep = ",")

#Q1
library(lattice)
png("scatterplott-matrix.png")
splom(dados[c(1,2,3,4)], groups=dados$Classe)
dev.off()

#Q2.a
setosa=subset(dados,dados$Classe=="Iris-setosa")
versicolor=subset(dados,dados$Classe=="Iris-versicolor")
virginica=subset(dados,dados$Classe=="Iris-virginica")

lim_x = xlim=c(min(dados$Tamanho_sépala),max(dados$Tamanho_sépala))
```

```
lim_y = ylim=c(min(dados$Largura_pétala),max(dados$Largura_pétala))

png("class_separation.png")
plot(setosa$Tamanho_sépala,setosa$Largura_pétala,xlim=lim_x,ylim=li
m_y,col="deepskyblue")
points(versicolor$Tamanho_sépala,versicolor$Largura_pétala,col="dee
ppink")
points(virginica$Tamanho_sépala,virginica$Largura_pétala,col="fores
tgreen")
# simple line separator: y=0x+0.75
line1<-data.frame(x=c(0,9),y=c(0.75,0.75))
lines(line1$x,line1$y,type='l')
# simple line separator: y=0x+1.55
line2<-data.frame(x=c(0,9),y=c(1.55,1.55))
lines(line2$x,line2$y,type='l')
dev.off()

#Q2.b

#Nessa questão basta traçar as FDPs das 3 classes para cada uma das
quatro características e analisar que aquelas cujos gráficos
estiverem mais distantes entre si devem ser as `melhores'
características.
```

```
iris<- read.table("plantas-iris.txt", header = TRUE, sep = ",")

#Q3. a e b - Falso positivo, falso negativo..... F-Measure

#esta é a função de uma reta no formato y=ax+b
f <- function(x,a,b) a*x+b

#vamos definir uma reta qualquer: e.g., 0.45x - 2.2
#digamos que esta reta divida o que é iris-setosa do resto. Vamos
testar todos os pontos para ver o acerto
#definamos as variaveis de erro e acerto para contar quantas vezes
se erra
#este cálculo está sendo realizado pra IRIS-SETOSA x DEMAIS, usando
as características 1 e 2. O mesmo precisa ser repetido para as
outras duas classes com as características escolhidas pelo aluno!

carac1 = 1
carac2 = 2
classe = 5
```

```
FP =0
VP =0
FN =0
VN =0
for(i in 1:dim(iris)[1]){
     y = f(iris[i,carac1],0.45,-2.2)
     if(y > iris[i,carac2]) {
          #abaixo da reta, logo não é iris-setosa
          if(iris[i,classe]=="Iris-setosa"){
               #disse que não era setosa, mas era => Falso
Negativo!
               FN <- FN + 1
          }else{
               #disse que não era setosa e não era mesmo =>
Verdadeiro negativo!
               VN <- VN + 1
          }
     }else{
          #acima da reta quer dizer que é Iris-setosa
          if(iris[i,classe]=="Iris-setosa"){
               #disse que era setosa... e era! => verdadeiro
positivo!

               VP <- VP + 1
          }else{
               #Disse que era setosa, mas não era! => falso
positivo!
               FP <- FP + 1
          }


     }
}
precision = VP/(VP+FP)
recall = VP/(VP+FN)
F = 2*(precision*recall)/(precision+recall)
FPR = FP/(FP+VN)
TPR = VP/(VP+FN)
```

Q3. c - Código simples <= certamente os especialistas em R podem fazer algo muito melhor!

```
iris <- read.table("plantas-iris.txt", header = TRUE, sep = ",")

#calculo da matriz de confusão
#esta é a função de uma reta no formato y=ax+b
f <- function(x,a,b) a*x+b

#vamos definir duas reta: 0.45x +0.5 e 0.45x +0.1
#digamos que uma reta divida o que é iris-setosa do resto e a outra
divide o que é Iris-virginica do resto. Vamos testar todos os
pontos para ver o acerto para estas duas retas
#este cálculo está sendo realizado usando as características 1 e 2.
O mesmo deve ser realizado com as características escolhidas pelo
aluno!

carac1 = 1
carac2 = 2
classe = 5


#definindo uma matriz para armazenar a confusão
confMat = matrix(nrow=3,ncol=3,c(0,0,0,0,0,0,0,0,0))
for(i in 1:dim(iris)[1]){
     y1 = f(iris[i,carac1],0.45,0.5)
     y2 = f(iris[i,carac1],0.45,0.1)
     if(y1 > iris[i,carac2]) {
          if(y2 > iris[i,carac2]) {
               #abaixo das duas retas, logo é iris-virgínica
               if(iris[i,classe]=="Iris-virginica"){
                    #Então está certo!
                    confMat[3,3]<-confMat[3,3] + 1
               }else{
                    #então está errado!
                    if(iris[i,classe ]=="Iris-setosa"){
                         confMat[3,1]<-confMat[3,1] + 1
                    }else{
                         confMat[3,2]<-confMat[3,2] + 1
                    }
               }
          }else{
               #entre as retas, logo é iris-versicolor
```

```
                if(iris[i,classe ]=="Iris-versicolor"){
                      #Então está certo!
                      confMat[2,2]<-confMat[2,2] + 1
                }else{
                      #então está errado!
                      if(iris[i,classe]=="Iris-setosa"){
                            confMat[2,1]<-confMat[2,1] + 1
                      }else{
                            confMat[2,3]<-confMat[2,3] + 1
                      }
                }
          }

     }else{
          #acima da reta quer dizer que é Iris-setosa
          if(iris[i,classe]=="Iris-setosa"){
                      #Então está certo!
                      confMat[1,1]<-confMat[1,1] + 1
                }else{
                      #então está errado!
                      if(iris[i,classe]=="Iris-versicolor"){
                            confMat[1,2]<-confMat[1,2] + 1
                      }else{
                            confMat[1,3]<-confMat[1,3] + 1
                      }
                }


     }
}
#P.S.: As medidas FP, FN, etc., podem ser calculadas utilizando-se
apenas a matriz de confusão, mas deve-se indicar qual classe é a
classe "positiva", sendo as demais "negativas".
```

4 -

Para responder a esta questão, é necessário ter compreendido os conceitos ilustrados nos slides 14-16 da Aula 1 de PRNN.

As três classes de problema pedidas nesta questão foram: Classificação, Regressão e Agrupamento.

Basicamente, para um problema de **Classificação**, é preciso ter-se Classes para as quais serão mapeadas as características existentes. Se não houverem possíveis

classes, não há um problema de classificação. Possíveis classes são dados categóricos e limitados de modo que para cada instância exista uma associação a exatamente uma classe.
Por sua vez, um problema de **Regressão** é similar a um problema de classificação, porém a "classe" precisa ser um valor numérico contínuo (e.g., entre 0 e 1).

Por fim, para um problema de **agrupamento**, podem ou não existirem classes pré-definidas, estas classes podem ser utilizadas para análise do agrupamento. Uma vez que possui poucos requisitos, praticamente qualquer base de dados pode ser utilizada para um problema de agrupamento, apesar de que não há garantias que o agrupamento resultante seja pertinente.

Para problemas de classificação e regressão, em PR normalmente se usam dados *quantitativos*, os quais são utilizados para calcular a relação entre os padrões. Quando se dispões de dados *qualitativos*, pode-se efetuar a conversão para dados numéricos - por exemplo, o estado AC, pode ser 0, AM pode ser 1, AP pode ser 2, e assim por diante. Na prática pode-se fazer uma classificação com estes valores. Entretanto a escala de valores tende a influenciar no resultado. No exemplo acima, os resultados poderiam fazer sentido apenas se existisse uma relação entre o problema estudado e a ordem da geração das classes. (e.g., AP valesse o dobro de AM). Por isso, ao utilizar o Weka é necessário muito cuidado, pois tal ferramenta realiza esta conversão automaticamente, o que pode levar a resultados inconsistentes!

Para termos um gabarito, será considerado o caso ideal - ou seja, os dados serão avaliados levando-se em consideração o seu significado, mesmo sabendo que na prática a ferramenta Weka poderia realizar a classificação.

Em Fundamentos em Analytics, foram utilizadas as bases: Salários de TI, Atividades das Máquinas do DSC, Ask Ubuntu, Assiduidade dos deputados, Velocidade do ISP e a base do TSE.

Como dito anteriormente, para se ter um problema de classificação, é preciso que haja uma classe. Em nenhuma das bases foi definida explicitamente uma classe, entretanto caso alguma das características puder ser tal classe, a base será considerada como sendo para classificação.

Para se ter um problema de regressão, é preciso ter uma característica que possa ser a 'classe numérica'. Além disso, é preciso, no mínimo, uma segunda característica numérica para calcular o relacionamento. Caso não haja uma característica numérica, mas haja uma categórica que possa ser traduzida

automaticamente em números (ex.: muito bom, bom, ruim), poderá ser considerada para regressão.

Todas as bases de dados podem ser utilizados para agrupamento. O único entrave é para os algoritmos de agrupamento que utilizam a classe como medida de desempenho. Para esses, só podem ser utilizadas as bases de dado que possuem classes.

Salários de TI - Esta base tem por características: Cidade, UF, Salário Bruto, Horas Diárias, Tempo de Empresa, Experiência Profissional, Iniciativa privada / concursado, Cargo, Formação, Pós-Graduação ou Certificação.

Nesta base, apesar de não ter sido formalmente definida nenhuma classe, poderia-se ter definido uma classe com base nos dados existentes, e.g., Iniciativa privada ou concursado (é possível identificar se um profissional é da iniciativa privada com o restante dos dados?) e Pós-Graduação (é possível, por intermédio dos outros dados, classificar um funcionário em pós-graduado ou não?).

Também Poderia-se utilizar alguns dados numéricos para um problema de regressão, e.g., salário (é possível estimar o salário de um funcionário com base nas demais características?) e horas diárias (é possível identificar quantas horas um funcionário trabalha com base nas demais informações?).

Portanto, Salários de TI poderia ser utilizado para os três tipos de problema.

Atividades das máquinas do DSC - Esta base tem as características: intervalo, se ociosa, maquina e laboratorio.

Também não foi definida nesta base nenhuma classe. Entretanto, tanto ociosa quanto laboratório poderiam ser utilizados para definir a classe. (*Nota: Na prática, a quantidade de características é muito pequena para um problema real. Mas na teoria é possível. Também não faria muito sentido ter máquina como característica para identificar a classe, já que no nome da máquina é possível identificar o laboratório!*). Rigorosamente falando, o Intervalo também poderia ser utilizado para um problema de regressão. Na prática, não iria funcionar bem, uma vez que seria preciso estimar um dado numérico a partir de dados categóricos, a não ser que os dados categóricos representassem uma escala de valor, o que não é o caso.

Portanto, Atividades das máquinas do DSC poderia ser utilizado para um problema de classificação e agrupamento. Mas não recomenda-se para regressão.

Ask ubuntu - display name, reputation, views, up votes, down votes.

Neste problema, nenhuma característica poderia ser utilizada como classe, pois não existe uma característica categórica e nem uma com valores numéricos limitados (ex.: entre 0 e 2, onde cada valor poderia representar uma classe).

Entretanto, poderia ser utilizado para um problema de regressão (provavelmente, calculando a reputação a partir dos demais dados).

Portanto a base ask ubuntu, poderia ser utilizada para um problema de regressão e agrupamento, mas não de classificação.

Assiduidade dos deputados - Nome, estado, região, partido, posição, gastos total, presenças total

Algumas características podem ser utilizadas para definir as classes: estado, região, partido e posição. Também podem ser utilizadas para regressão as características de gastos e presença. Similarmente ao problema das máquinas do DSC, precisaríamos de mais dados numéricos para uma regressão consistente. Entretanto, caso haja uma relação entre gastos e presença, a estimativa de um pode ser feita em função do outro.

Portanto, a base de assiduidade poderia ser utilizada para problemas de classificação, regressão e agrupamento.

Velocidade do ISP - Ano, mes, dia, hora, velocidade, provedor, estado

Geralmente dados temporais podem atrapalhar um pouco a classificação devido ao viés gerado por sazonalidades. Por exemplo, fica difícil para a rede generalizar dados para o restante do ano, uma vez que esta foi treinada com dados de dezembro. Para tanto, existem as Séries Temporais (que serão estudadas mais adiantes).

Daria, entretanto, para utilizar o provedor ou o estado como classes. Para um problema de regressão, entretanto não seria possível, pois não seria possível estimar os valores a partir de dados meramente categóricos.

Portanto, a base de velocidade do ISP poderia ser utilizada para problemas de classificação e agrupamento.

Base do TSE - Como são diversas bases, não vamos descrevê-la, mas a essência é a mesma que já vem sendo descrita. Mas certamente o resultado da eleição pode ser considerada para um problema de classificação e a quantidade de votos pode ser utilizada para regressão. Portanto, a base pode ser utilizada para os três tipos de problema.

E quais modificações seriam necessárias para permitir que todas as bases pudessem ser usadas em problemas de (a) classificação e (b) regressão.

(a)

askubuntu: Aqui seria necessário definir classes com base nos valores de alguma característica, por exemplo, a reputação.

(b)

Atividade das máquinas do DSC: Neste caso, seria necessário ter pelo menos uma outra característica numérica.

Velocidade do ISP: Neste caso, seria necessário ter pelo menos uma outra característica numérica.