Thema09_EDA

Introduction

Pancreatic ductal adenocarcinoma (PDAC) is one of the deadiest cancers. The chances of survival are increased when diagnosed in an early stage. However, PDAC shows symptoms when it already spread throughout the body. Most of the time, it's too late by then. There may be a way to detect PCAD in an early stage with a simple urine test, with the use of the following biomarkers: creatinine (Urinary biomarker of kidney function) LYVE1 (Urinary levels of Lymphatic vessel endothelial hyaluronan receptor 1, a protein that may play a role in tumor metastasis), REG1A and REG1B (Urinary levels of a protein that may be associated with pancreas regeneration.), and TFF1 (Urinary levels of Trefoil Factor 1, which may be related to regeneration and repair of the urinary tract)

the attributes in the data interesting for this research are the biomarker values mentioned before. There is also an attribute called diagnosis, in which the diagnosis of the sample is stated, where 1 means no PDAC, 2 means benign hepatobiliary disease (non cancerous, non harmful pancreatic condition), and 3 means that the sample has PDAC.

Research question: Is it possible to detect pancreatic cancer using values of the urinary biomarkers?

EDA

Codebook

```
myData <- read.csv("Data/Debernardi et al 2020 data.csv")

columns <- colnames(myData)
type <- c("character", "character", "character", "double", "character", "double", "logical", "logical",
unit <- c(NA, NA, NA, "years", "F/M", NA, NA, NA, "U/ml", "mg/ml", "ng/ml", "ng/ml", "ng/ml", "ng/ml")
descriptions = c("Unique string identifying each subject", "Cohort 1, previously used samples; Cohort 2
codebook <- data.frame(columns, type, unit, descriptions)
write.csv(codebook, "Codebook.csv", row.names = FALSE)
knitr::kable(codebook, caption="Table 1: The Codebook", format = 'latex') %>%
  kable_styling(full_width = F) %>%
  column_spec(1, bold = T) %>%
  column_spec(4, width = "22em")
```

Data exploration

Visualization

To look at the data and any possible missing data, here's a table containing only the relevant columns for this research.

```
relevant <- myData[c(4, 6, 9:14)]
knitr::kable(relevant, caption="Table 2: Values of biomarkers and the diagnosis")
```

Table 1: Table 1: The Codebook

columns	type	unit	descriptions
sample_id	character	NA	Unique string identifying each subject
patient_cohort	character	NA	Cohort 1, previously used samples; Cohort 2, newly added samples
sample_origin	character	NA	BPTB: Barts Pancreas Tissue Bank, London, UK; ESP: Spanish National Cancer Research Centre, Madrid, Spain; LIV: Liverpool University, UK; UCL: University College London, UK
age	double	years	Age in years
sex	character	F/M	M = male, F = female
diagnosis	double	NA	1 = control (no pancreatic disease), 2 = benign hepatobiliary disease (119 of which are chronic pancreatitis); 3 = Pancreatic ductal adenocarcinoma, i.e. pancreatic cancer
stage	logical	NA	For those with pancratic cancer, what stage was it? One of IA, IB, IIA, IIIB, III, IV
benign_sample_diagnosis	logical	NA	For those with a benign, non-cancerous diagnosis, what was the diagnosis?
plasma_CA19_9	double	U/ml	Blood plasma levels of CA 19–9 monoclonal antibody that is often elevated in patients with pancreatic cancer. Only assessed in 350 patients (one goal of the study was to compare various CA 19-9 cutpoints from a blood sample to the model developed using urinary samples).
creatinine	double	mg/ml	Urinary biomarker of kidney function
LYVE1	double	ng/ml	Urinary levels of Lymphatic vessel endothelial hyaluronan receptor 1, a protein that may play a role in tumor metastasis
REG1B	double	ng/ml	Urinary levels of a protein that may be associated with pancreas regeneration.
TFF1	double	ng/ml	Urinary levels of Trefoil Factor 1, which may be related to regeneration and repair of the urinary tract
REG1A	double	ng/ml	Urinary levels of a protein that may be associated with pancreas regeneration. Only assessed in 306 patients (one goal of the study was to assess REG1B vs REG1A)

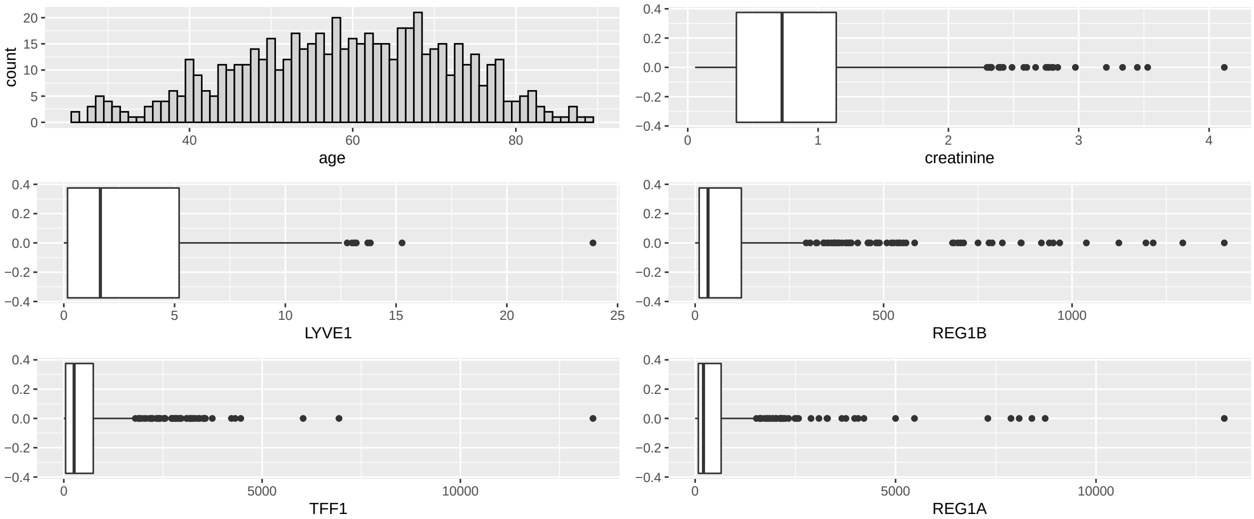
Table 2: Table 2: Values of biomarkers and the diagnosis

age	diagnosis	plasma_CA19_9	creatinine	LYVE1	REG1B	TFF1	REG1A
33	1	1.170000e+01	1.832220	0.8932192	52.9488400	6.542822e+02	1262.000
81	1	NA	0.972660	2.0375850	94.4670300	2.094882e+02	228.407
51	1	7.000000e+00	0.780390	0.1455889	102.3660000	4.611410e+02	NA
61	1	8.000000e+00	0.701220	0.0028049	60.5790000	1.429500e+02	NA
62	1	9.000000e+00	0.214890	0.0008596	65.5400000	4.108800e+01	NA
53	1	NA	0.848250	0.0033930	62.1260000	5.979300e+01	NA
70	1	NA	0.622050	0.1743808	152.2770000	1.175160e+02	NA
58	1	1.100000e+01	0.893490	0.0035740	3.7300000	4.029400e+01	NA
59	1	NA	0.486330	0.0019453	7.0210000	2.678200e+01	NA
56	1	2.400000e+01	0.610740	0.2787785	83.9280000	1.918500e+01	NA
77	1	NA	0.294060	0.0011762	6.2180000	2.829700e+01	NA
71	1	2.300000e+01	1.051830	0.8603368	243.0820000	6.082840e+02	NA
49	1	NA	0.859560	1.4163140	151.8307700	7.418990e+01	505.571
53	1	7.000000e+00	1.911390	1.5167730	150.8900000	5.906860e+02	NA
56	1	1.200000e+01	0.916110	0.5996449	93.8110000	9.357600e+01	NA
60	1	2.800000e+01	0.508950	0.0020358	24.3660000	1.969800e+01	NA
69	1	9.000000e+00	0.418470	0.0016739	17.1020000	3.264070e-02	NA
60	1	4.700000e+01	0.803010	0.0032120	3.5880000	3.007100e+01	NA
55	1	1.700000e+01	1.289340	2.2853510	67.4680000	2.698050e+02	NA
28	1	8.700000e+00	0.508950	0.5830097	13.6190600	2.671935e+02	381.000
54	1	NA	1.244100	0.0049764	5.5073501	1.931457e+02	113.000
50	1	8.700000e+00	0.950040	0.0038002	56.3991330	1.922589e+02	137.000
40	1	NA	0.769080	0.6539844	14.6075790	3.412675e+02	NA
74	1	NA	0.316680	0.5830097	25.5203500	1.465886e+02	111.531
63	1	NA	0.757770	2.4401800	21.2294800	1.094211e+02	903.000
50	1	NA	0.780390	1.0444110	7.3556560	2.503443e+02	149.000
47	1	NA	0.950040	0.7486172	19.0209880	2.485707e+02	736.000
45	1	9.600000e+00	1.357200	2.3928640	28.5092850	3.536567e+02	563.000
35	1	4.000000e+00	0.248820	0.0009953	9.2451660	6.030701e+00	624.000
30	1	1.080000e+01	1.187550	1.6003130	22.4712810	2.991184e+02	570.000
48	1	NA	0.723840	0.0028954	5.3540000	4.334300e+01	NA
44	1	NA	1.221480	5.3091970	21.9650000	9.715530e+02	NA
44	1	NA	1.176240	4.2393080	7.6200000	4.363610e+02	NA
56	1	NA	0.135720	0.0005429	2.2760000	1.058620e-02	NA
58	1	NA	1.210170	1.0085840	45.1856930	2.689671e+02	116.998
44	1	NA	2.024490	6.9252810	53.1100000	1.274490e+03	NA
48	1	NA	0.158340	0.0006334	1.2100000	3.526000e+00	NA
48	1	NA	0.339300	0.0013572	1.9070000	3.044000e+00	NA
41	1	NA	0.101790	1.0138240	2.7950000	1.820000e+00	NA
45	1	NA	0.859560	0.9638040	29.5220000	3.630760e+02	NA
45	1	NA	0.599430	4.0902390	15.3280000	6.771100e+01	NA
45	1	NA	0.588120	0.4626530	3.6120000	3.423700e+01	NA
48	1	NA	1.051830	7.3001920	118.3250000	1.424961e+03	NA
89	1	NA	0.644670	6.8368430	76.8340000	9.815450e+02	NA
87	1	5.647590e-02	0.067860	0.0002714	0.7303665	5.293100e-03	NA
50	1	NA	0.610740	0.1103106	16.4332700	1.745229e+02	103.184
66	1	4.060258e+00	0.757770	0.0089199	5.4198700	8.919911e+00	NA
59	1	9.085780e-01	1.911390	0.0621642	50.1122000	6.216421e+01	NA
59	1	3.389304e+00	1.266720	0.0071137	5.2258900	7.113679e+00	NA
36	1	5.128234e+00	0.961350	0.0194258	47.2546850	1.942577e+01	NA
70	1	1.135926e+00	0.395850	0.0127807	10.8874650	1.278068e+01	NA
63	1	1.938949e+00	0.429780	0.0017191	5.7994050	3.352280e-02	NA
63	1	2.059078e+00	0.327990	0.0013120	219.8605000	2.558320e-02	NA
59	1	5.161092e-01	0.882180	0.0084047	8.2321800	8.404655e+00	NA
67	1	1.847713e+00	0.384540	0.0016826	15.0611800	1.682586e+00	NA

As is it obvious to see, the REG1A column contains a lot of missing values. The question is exactly how is this possible and what does it mean for the rest of the data? The first and most simple solution is that we don't need this column at all, since the REG1B is similar in function and containing all of the data. Another solution that's not very logical, is to use only the rows where there is a value in the REG1A column. The most logical solution in my opinion is to look at every column separately, and then remove the missing data. When the cleaned data then is plotted, it's possible to look at all the columns together and see if there is any correlation or trend to figure out.

knowing this, let's have a look at the important raw data with the use of boxplots.

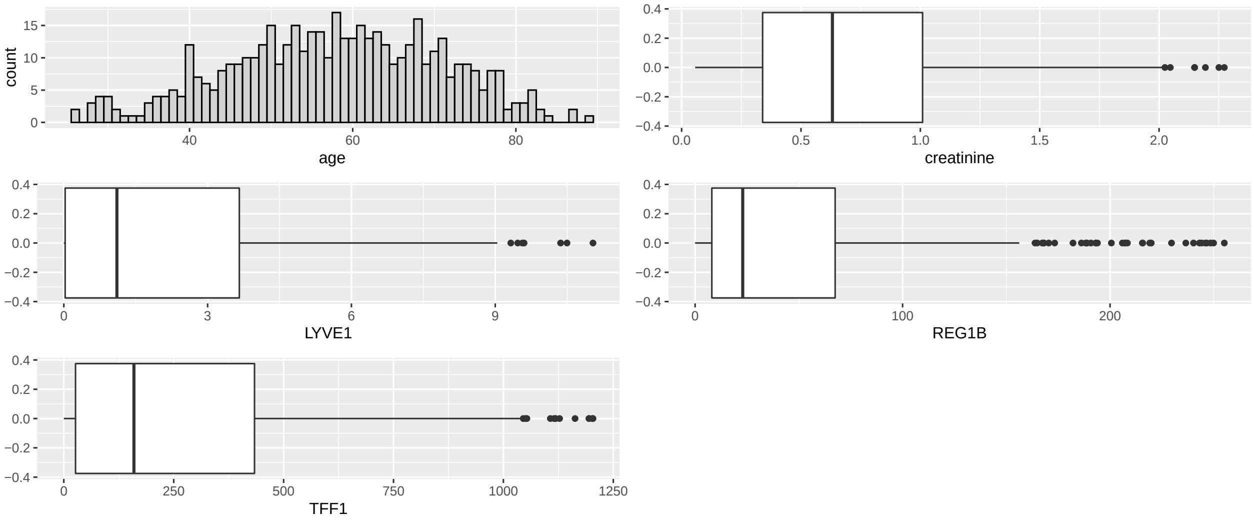
```
p1 <- ggplot(myData, aes(x=age)) +
  geom_histogram(fill = "lightgrey", col = "black", binwidth = 1)

p2 <- ggplot(myData, aes(x=creatinine)) +
  geom_boxplot()
p3 <- ggplot(myData, aes(x=LYVE1)) +
  geom_boxplot()
p4 <- ggplot(myData, aes(x=REG1B)) +
  geom_boxplot()
p5 <- ggplot(myData, aes(x=TFF1)) +
  geom_boxplot()
p6 <- ggplot(myData, aes(x=REG1A)) +
  geom_boxplot()

grid.arrange(p1, p2, p3, p4, p5, p6, nrow = 3)
```



as you can see, there are also a lot of outliers which we need to check. If we remove the outliers and clean up the data, without removing the missing values of REG1A, we'll get something like this.

```
Q <- quantile(myData$creatinine, probs=c(.25, .75), na.rm = FALSE)
iqr <- IQR(myData$creatinine)
up <- Q[2]+1.5*iqr
low <- Q[1]-1.5*iqr

myDataNO <- subset(myData, myData$creatinine > low & myData$creatinine < up)

Q <- quantile(myDataNO$LYVE1, probs=c(.25, .75), na.rm = FALSE)
iqr <- IQR(myDataNO$LYVE1)
up <- Q[2]+1.5*iqr
```

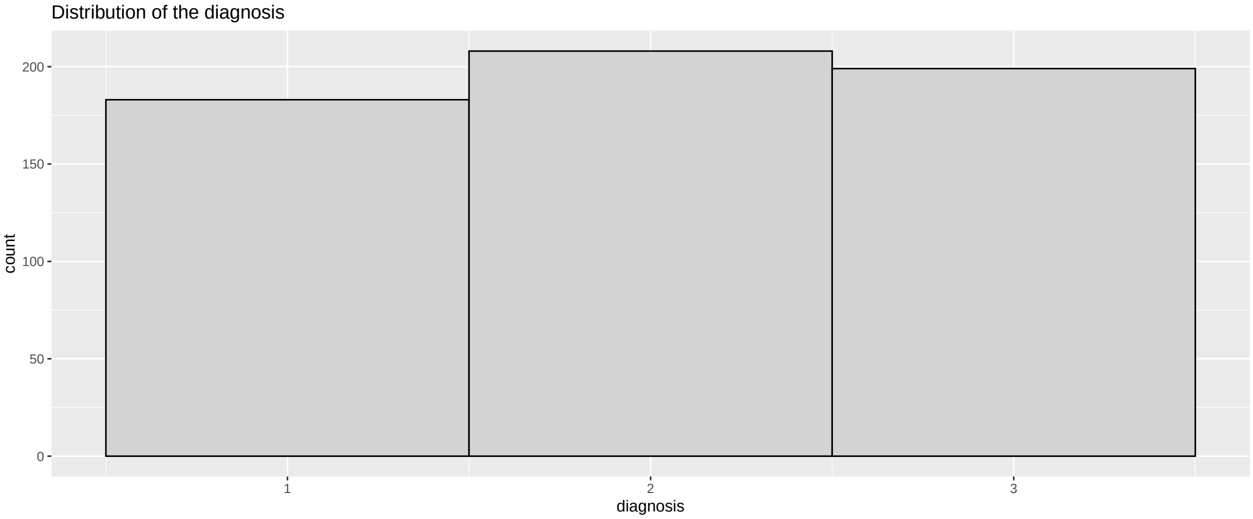
```
low <- Q[1]-1.5*iqr

myDataNO <- subset(myDataNO, myDataNO$LYVE1 > low & myDataNO$LYVE1 < up)

Q <- quantile(myDataNO$REG1B, probs=c(.25, .75), na.rm = FALSE)
iqr <- IQR(myDataNO$REG1B)
up <- Q[2]+1.5*iqr
low <- Q[1]-1.5*iqr

myDataNO <- subset(myDataNO, myDataNO$REG1B > low & myDataNO$REG1B < up)

Q <- quantile(myDataNO$TFF1, probs=c(.25, .75), na.rm = FALSE)
iqr <- IQR(myDataNO$TFF1)
up <- Q[2]+1.5*iqr
low <- Q[1]-1.5*iqr

myDataNO <- subset(myDataNO, myDataNO$TFF1 > low & myDataNO$TFF1 < up)

p1 <- ggplot(myDataNO, aes(x=age)) +
  geom_histogram(fill = "lightgrey", col = "black", binwidth = 1)

p2 <- ggplot(myDataNO, aes(x=creatinine)) +
  geom_boxplot()
p3 <- ggplot(myDataNO, aes(x=LYVE1)) +
  geom_boxplot()
p4 <- ggplot(myDataNO, aes(x=REG1B)) +
  geom_boxplot()
p5 <- ggplot(myDataNO, aes(x=TFF1)) +
  geom_boxplot()

grid.arrange(p1, p2, p3, p4, p5, nrow = 3)
```



This looks a lot cleaner, however it is not wise to remove these outliers as they may contain important data in a later stage.

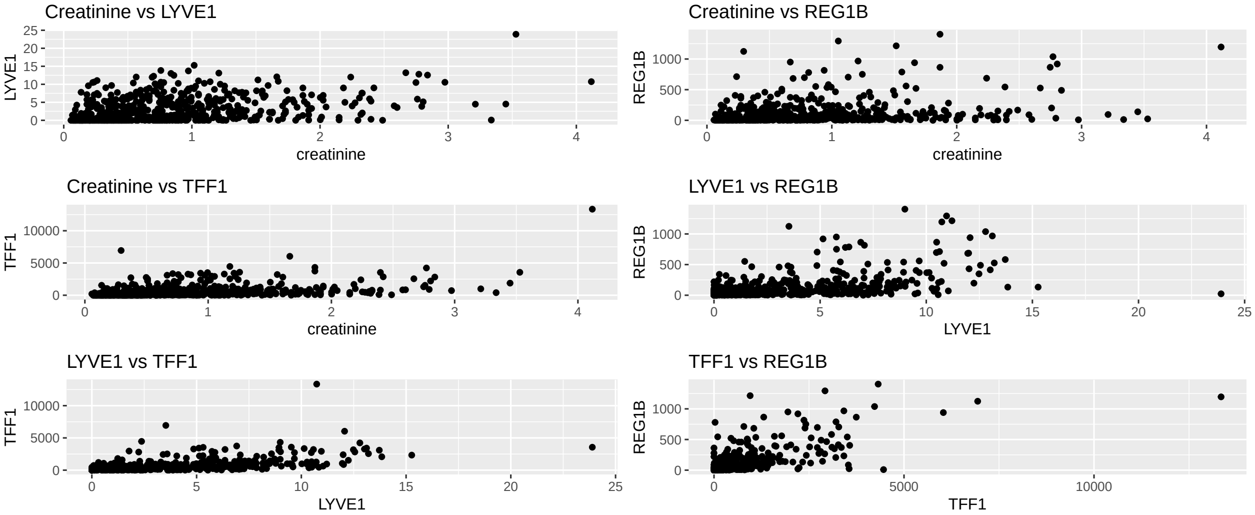
Before proceeding, it is quite helpful to look at the distribution of the diagnosis column, to see if they're evenly distributed or not.

```
ggplot(myData, aes(x=diagnosis)) +
  geom_histogram(fill = "lightgrey", col = "black", binwidth = 1) +
  ggtitle("Distribution of the diagnosis")
```



Now we can check if the different biomarkers correlate with each other in any way.

```
p1 <- ggplot(myData, aes(x=creatinine, y=LYVE1)) +
  geom_point() +
  ggtitle("Creatinine vs LYVE1")
p2 <- ggplot(myData, aes(x=creatinine, y=REG1B)) +
  geom_point() +
  ggtitle("Creatinine vs REG1B")
p3 <- ggplot(myData, aes(x=creatinine, y=TFF1)) +
  geom_point() +
  ggtitle("Creatinine vs TFF1")
p4 <- ggplot(myData, aes(x=LYVE1, y=REG1B)) +
  geom_point() +
  ggtitle("LYVE1 vs REG1B")
p5 <- ggplot(myData, aes(x=LYVE1, y=TFF1)) +
  geom_point() +
  ggtitle("LYVE1 vs TFF1")
p6 <- ggplot(myData, aes(x=TFF1, y=REG1B)) +
  geom_point() +
  ggtitle("TFF1 vs REG1B")

grid.arrange(p1, p2, p3, p4, p5, p6, nrow=3)
```

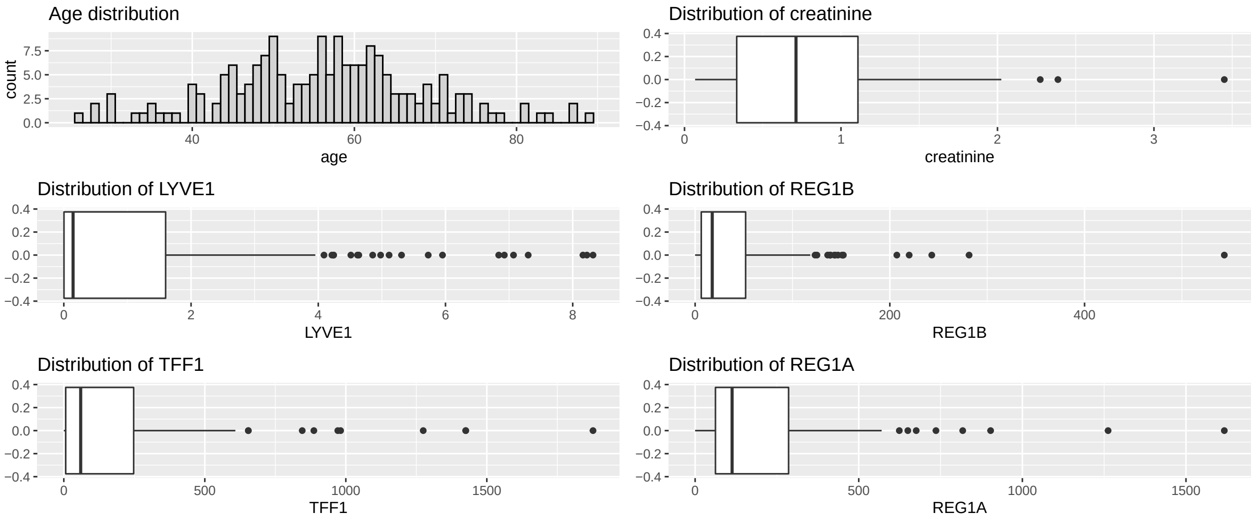


I think it's safe to assume that none of these correlate with each other in any way. One thing to notice however, is that LYVE1, REG1B and TFF1 have a lot of values close to zero. We can figure out if this is important or not in a future stage of the research.

To ensure that all the values are completely independant we can perform a principal component analysis.

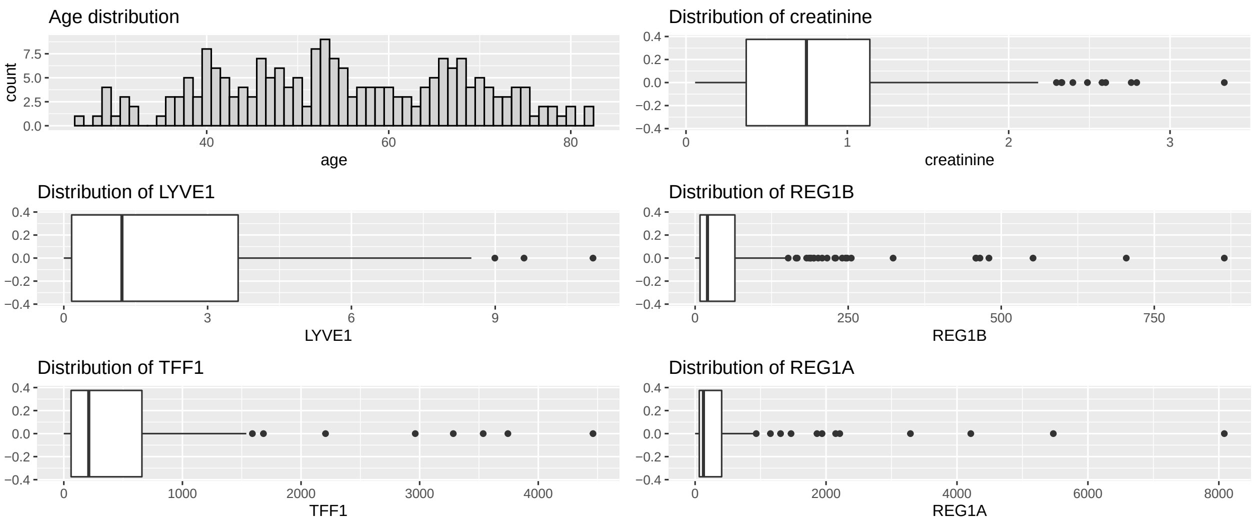
What we should do now, is look at the values of those urinary levels with each diagnosis and see if there are any obvious differences.
Let's start with the samples that do not have PDAC

```
myData1 <- subset(myData, myData$diagnosis == 1)

p1 <- ggplot(myData1, aes(x=age)) +
  geom_histogram(fill = "lightgrey", col = "black", binwidth = 1) +
  ggtitle("Age distribution")

p2 <- ggplot(myData1, aes(x=creatinine)) +
  geom_boxplot() +
  ggtitle("Distribution of creatinine")
p3 <- ggplot(myData1, aes(x=LYVE1)) +
  geom_boxplot() +
  ggtitle("Distribution of LYVE1")
p4 <- ggplot(myData1, aes(x=REG1B)) +
  geom_boxplot() +
  ggtitle("Distribution of REG1B")
p5 <- ggplot(myData1, aes(x=TFF1)) +
  geom_boxplot() +
  ggtitle("Distribution of TFF1")
p6 <- ggplot(myData1, aes(x=REG1A)) +
  geom_boxplot() +
  ggtitle("Distribution of REG1A")

grid.arrange(p1, p2, p3, p4, p5, p6, nrow = 3)
```



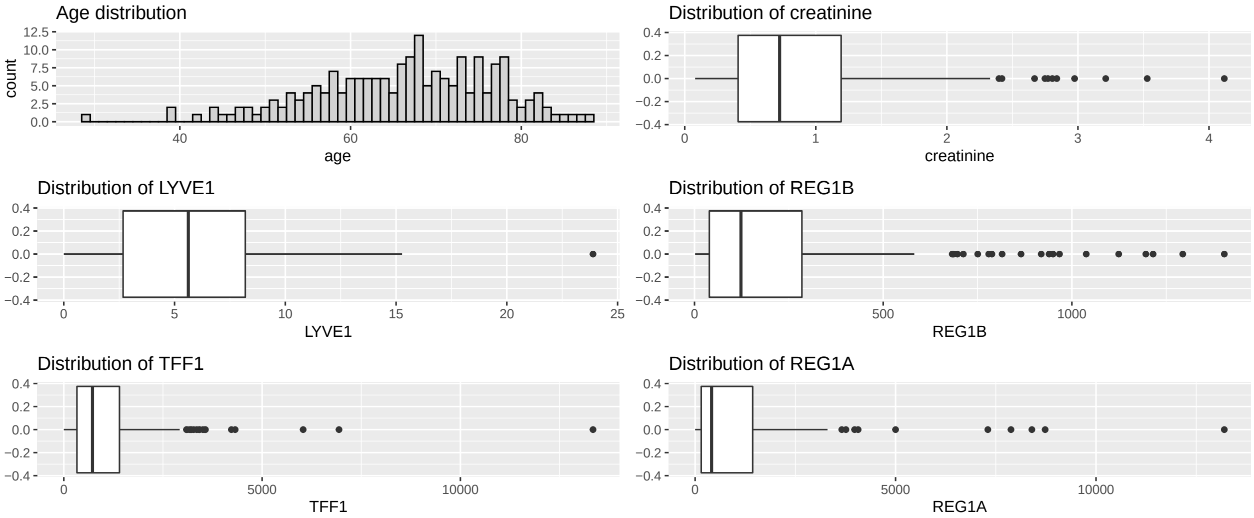
Here are the values of the samples with non-cancerous pancreatic conditions.

```
myData2 <- subset(myData, myData$diagnosis == 2)

p1 <- ggplot(myData2, aes(x=age)) +
  geom_histogram(fill = "lightgrey", col = "black", binwidth = 1) +
  ggtitle("Age distribution")

p2 <- ggplot(myData2, aes(x=creatinine)) +
  geom_boxplot() +
  ggtitle("Distribution of creatinine")
p3 <- ggplot(myData2, aes(x=LYVE1)) +
  geom_boxplot() +
  ggtitle("Distribution of LYVE1")
p4 <- ggplot(myData2, aes(x=REG1B)) +
  geom_boxplot() +
  ggtitle("Distribution of REG1B")
p5 <- ggplot(myData2, aes(x=TFF1)) +
  geom_boxplot() +
  ggtitle("Distribution of TFF1")
p6 <- ggplot(myData2, aes(x=REG1A)) +
  geom_boxplot() +
  ggtitle("Distribution of REG1A")

grid.arrange(p1, p2, p3, p4, p5, p6, nrow = 3)
```



And finally the samples that do have PDAC.

```
myData3 <- subset(myData, myData$diagnosis == 3)

p1 <- ggplot(myData3, aes(x=age)) +
  geom_histogram(fill = "lightgrey", col = "black", binwidth = 1) +
  ggtitle("Age distribution")

p2 <- ggplot(myData3, aes(x=creatinine)) +
  geom_boxplot() +
  ggtitle("Distribution of creatinine")
p3 <- ggplot(myData3, aes(x=LYVE1)) +
  geom_boxplot() +
  ggtitle("Distribution of LYVE1")
p4 <- ggplot(myData3, aes(x=REG1B)) +
  geom_boxplot() +
  ggtitle("Distribution of REG1B")
p5 <- ggplot(myData3, aes(x=TFF1)) +
  geom_boxplot() +
  ggtitle("Distribution of TFF1")
p6 <- ggplot(myData3, aes(x=REG1A)) +
  geom_boxplot() +
  ggtitle("Distribution of REG1A")

grid.arrange(p1, p2, p3, p4, p5, p6, nrow = 3)
```



At first glance, there are notable differences but we can further investigate this in future steps of this research.

```
write.csv(myData, "Data/DataCleaned.csv")
```