# Load files into R, then Split the first column to extract the patient ID and time of sample obtained

split\_Biomarker <- strsplit(data$Biomarker, "-")

data$Patient\_ID <- sapply(split\_Biomarker, function(x) x[1])

data$Time\_Taken <- sapply(split\_Biomarker, function(x) x[2])

# Filter the data for 0 weeks and 6 weeks, excluding 12 months

data\_filtered <- data[data$Time\_Taken %in% c("0weeks", "6weeks"), ]

biomarker\_columns <- c("IL-8", "VEGF-A", "OPG", "TGF-beta-1", "IL-6", "CXCL9", "CXCL1", "IL-18", "CSF-1")

# Check which patients have data from both 0 and 6 weeks; exclude the ones that have data only from one timepoint

not\_included\_twice <- data\_filtered$Patient\_ID[duplicated(data\_filtered$Patient\_ID) | duplicated(data\_filtered$Patient\_ID, fromLast = TRUE)] #3 patients removed

data\_filtered\_duplicated <- data\_filtered[data\_filtered$Patient\_ID %in% not\_included\_twice, ]

# Subset the data for the current Patient ID and Time\_Taken

patient\_data\_0weeks <- data\_filtered\_duplicated[data\_filtered\_duplicated$Time\_Taken == "0weeks",]

patient\_data\_6weeks <- data\_filtered\_duplicated[data\_filtered\_duplicated$Time\_Taken == "6weeks",]

# List of biomarkers for which you want to perform the Shapiro-Wilk test

biomarkers <- c("IL-8", "VEGF-A", "OPG", "TGF-beta-1", "IL-6", "CXCL9", "CXCL1", "IL-18", "CSF-1")

# Loop through column between 2 and 10 and get their names and perform the Shapiro-Wilk test on the values of each column each time

for (i in 2:10) { for (col\_name in names(data\_filtered\_duplicated)[i]) {column\_data <- data\_filtered\_duplicated[[col\_name]]

result <- shapiro.test(column\_data)

shapiro\_results[[col\_name]] <- result}}

# Create a list to store the test results

shapiro\_results <- list()

//

# Create new dataframe

data\_filtered\_duplicated\_v2 <- as.data.frame(data\_filtered\_duplicated)

# Save as new result

results <- data.frame(test=colnames(data\_filtered\_duplicated\_v2)[c(2:8,10)], p = NaN, stringsAsFactors=F)

n=0

# Perform Wilcoxon signed-rank test for the non-normally distributed biomarkers

for (i in c(2:8, 10)){

n=(n+1)

results$p[n] <- wilcox.test(data\_filtered\_duplicated\_v2[data\_filtered\_duplicated\_v2$Time\_Taken == "0weeks", i], data\_filtered\_duplicated\_v2[data\_filtered\_duplicated\_v2$Time\_Taken == "6weeks", i])$p.value}

results =

test p

1 IL-8 0.006802253

2 VEGF-A 0.014374517

3 OPG 0.282226020

4 TGF-beta-1 0.023917771

5 IL-6 0.730938242

6 CXCL9 0.077568304

7 CXCL1 0.010181674

8 CSF-1 0.069414155

# Perform paired t-test for normally distributed biomarker

t\_test\_result <- t.test(patient\_data\_0weeks$`IL-18`, patient\_data\_6weeks$`IL-18`, paired = TRUE)

//

# Calculate the probability of making at least one type I error

# Set the desired overall significance level

alpha <- 0.05 # For example, a 5% overall significance level

# Define the number of independent tests (assumed to have true null hypotheses)

N <- 115

# Calculate the Bonferroni-corrected significance level

alpha\_bonferroni <- alpha / 115

# Calculate the probability of making at least one Type I error

prob\_at\_least\_one\_error <- 1 - (1 - alpha\_bonferroni)^115

cat("Probability of making at least one Type I error:", prob\_at\_least\_one\_error, "\115")

Probability of making at least one Type I error: 0.04878092

//

# Re-do Wilcoxon test but with Bonferroni correction

results$padj <- p.adjust(results$p, method = "bonferroni")

results=

test p padj

1 IL-8 0.006802253 0.05441802

2 VEGF-A 0.014374517 0.11499613

3 OPG 0.282226020 1.00000000

4 TGF-beta-1 0.023917771 0.19134216

5 IL-6 0.730938242 1.00000000

6 CXCL9 0.077568304 0.62054643

7 CXCL1 0.010181674 0.08145340

8 CSF-1 0.069414155 0.55531324

//

# Load in and read covariates + biomarkers files. Then do strsplit to split Patient\_ID and weeks

biomarkers\_unfiltered$PatientID <- unlist(strsplit(biomarkers\_unfiltered$Biomarker, "-"))[seq(1, nrow(biomarkers\_unfiltered)\*2,2)]

biomarkers\_unfiltered$Weeks <- unlist(strsplit(biomarkers\_unfiltered$Biomarker, "-"))[seq(2, nrow(biomarkers\_unfiltered)\*2,2)]

subset <- biomarkers\_unfiltered[biomarkers\_unfiltered$Weeks == "0weeks",]

# Merge the two files: biomarkers & covariates on Patient\_ID

merged <- merge(biomarkers\_unfiltered, covariates)

subset <- merged[merged$Weeks == "0weeks",]

# Convert male/female category from ‘1 or 2’

subset$sex\_category <- factor(c("male", "female")[subset$`Sex (1=male, 2=female)`])

# Convert smoker category from ‘1 or 2’

subset$smoker\_category <- factor(c("yes", "no")[subset$`Smoker (1=yes, 2=no)`])

# Save model as ‘tab’ and pick at random 3 biomarkers for explanatory variables, along with sex and smoking status.

tab <- subset[, c("IL-6","VEGF-A", "CXCL9","smoker\_category", "sex\_category", "VAS-at-inclusion","Vas-12months")]

# Fit our model using 80% of patients at random

rows <- sample(1:nrow(tab), round(0.8\*nrow(tab)))

tab1 <- tab[rows,]

tab2 <- tab[-rows,]

fit <- lm(`Vas-12months`~., data=tab1)

# Model results =

Residuals:

Min 1Q Median 3Q Max

-6.5955 -1.9594 -0.2996 2.0075 5.9186

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -4.9233 5.6936 -0.865 0.38963

`IL-6` 0.9768 0.3026 3.229 0.00177 \*\*

`VEGF-A` 0.4734 0.4775 0.991 0.32431

CXCL9 -0.3838 0.3332 -1.152 0.25263

smoker\_categoryyes 0.4205 0.6803 0.618 0.53811

sex\_categorymale 0.3613 0.6356 0.568 0.57122

`VAS-at-inclusion` 0.3515 0.1144 3.071 0.00287 \*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.806 on 85 degrees of freedom

(2 observations deleted due to missingness)

Multiple R-squared: 0.2507, Adjusted R-squared: 0.1978

F-statistic: 4.739 on 6 and 85 DF, p-value: 0.0003289

# Predict VAS for the other 20% of patients, plot graph and do a correlation test

tab2$prediction <- predict(fit,tab2)

plot(tab2$`Vas-12months`,tab2$prediction)

cor.test(tab2$`Vas-12months`,tab2$prediction)

# Correlation results =

Pearson's product-moment correlation

data: tab2$`VAS-at-inclusion` and tab2$prediction

t = 5.6976, df = 21, p-value = 1.178e-05

alternative hypothesis: true correlation is not equal to 0

95 percent confidence interval:

0.5407009 0.9017801

sample estimates:

cor = 0.7792298