Idea de tesis

Martín Gabriel Cargnel

Resumen

Me interesa estudiar alternativas al modelo lineal cuando se busca explicar el efecto de distintas covariables en la variable de respuesta. La motivación surge de que los modelos lineales suelen ser la principal elección cuando se busca interpretabilidad, dado que muchos modelos de aprendizaje automático se consideran cajas negras. Es decir, que tienen mucha flexibilidad y poder predictivo pero son difíciles de interpretar por su complejidad. Sin embargo, existen técnicas de aprendizaje automático interpretable que permiten entender cómo el modelo utiliza las covariables.

Idealmente me gustaría encontrar técnicas que den siginificatividad estadística a modelos de aprendizaje automático y testearlas sobre un conjunto de datos para comparar sus resultados con el modelo lineal.

Organización

- Introducción
- Predecir ó interpretar: diferencias entre ambos objetivos y los modelos más utilizados en cada caso.
- Modelo lineal, supuestos y limitaciones: por qué es tan usado y algunas de sus limitaciones.
- Árboles y ensambles: explicar el algoritmo para crear árboles de decisión, luego bagging para terminar con Random Forest.
- Aprendizaje automático interpretable: un repaso de estas técnicas, sobre todo para modelos de árboles y cómo se puede obtener información similar a la que aporta un modelo lineal. A priori mencionaría importancias, PDP.
- Aplicación: Comparar modelos lineales vs. Random Forest con técnicas de aprendizaje automático interpretable.
- Discusión y futuros trabajos.
- Conclusiones

Ejemplo básico de aplicación

La idea de este ejemplo es comparar un ajuste lineal con un Random Forest combinado con técnicas de aprendizaje automático interpretable. Por lo que utilicé los datos de Diabetes, con el objetivo de entender el efecto de distintas variables en la progresión de la diabetes. El dataset cuenta con 442 observaciones y 12 variables, sin datos faltantes.

Análisis exploratorio

En primer lugar importo los datos y describo las variables en la siguiente tabla.

```
df_diabetes = read_csv("diabetes2.csv", show_col_types = FALSE)
```

Variable	Descripción
Id	Identificador único del paciente
AGE	edad en años del paciente
SEX	Sexo del paciente
BMI	Índice de masa corporal
BP	Presión arterial promedio
TC	Colesterol sérico total (s1)
LDL	Colesterol "malo" (s2)
HDL	Colesterol "bueno" (s3)
TCH	Colesterol total / HDL (s4)
LTG	logaritmo base 10 del nivel de triglicéridos (s5)
GLU	Nivel de azúcar en sangre (s6)
Y	Índice de progresión de la diabetes.

En la Figura 1 se ve un heatmap con la correlación entre las variables donde notamos que TC tiene una correlación muy alta con LDL (Figura 2a), también se ve que TCH y HDL tienen una correlación fuerte (Figura 2b). Entiendo que esto último se debe a que TCH calculó utilizando TC y HDL. Es por eso que voy a eliminar esas variables para evitar incluir variables muy correlacionadas o calculadas en base a otras.

```
cor_matrix <- cor(df_diabetes %>% select(-c("sex", "Id")),
                  use = "complete.obs")

melted_cor_matrix <- reshape2::melt(cor_matrix)

ggplot(data = melted_cor_matrix, aes(x = Var1, y = Var2, fill = value)) +
```

```
  geom_tile() +
  scale_fill_gradient2(low = "blue", high = "red", mid = "white",
                       midpoint = 0, limit = c(-1,1), space = "Lab",
                       name="Correlación") +
  geom_text(aes(label = round(value, 2)), color = "black", size = 4) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, vjust = 1,
                                   size = 12, hjust = 1)) +
  labs(x="", y="")
```

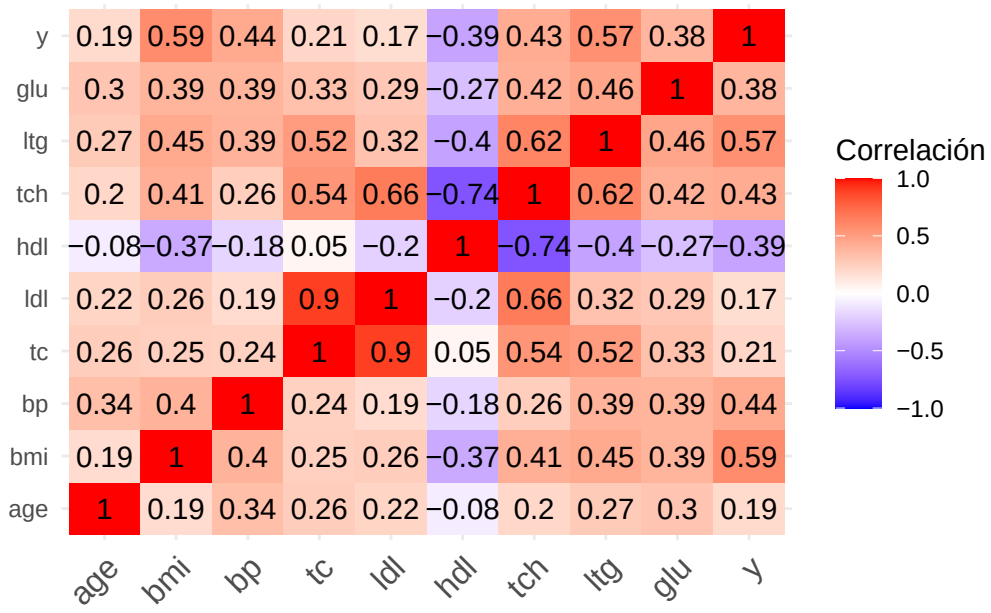
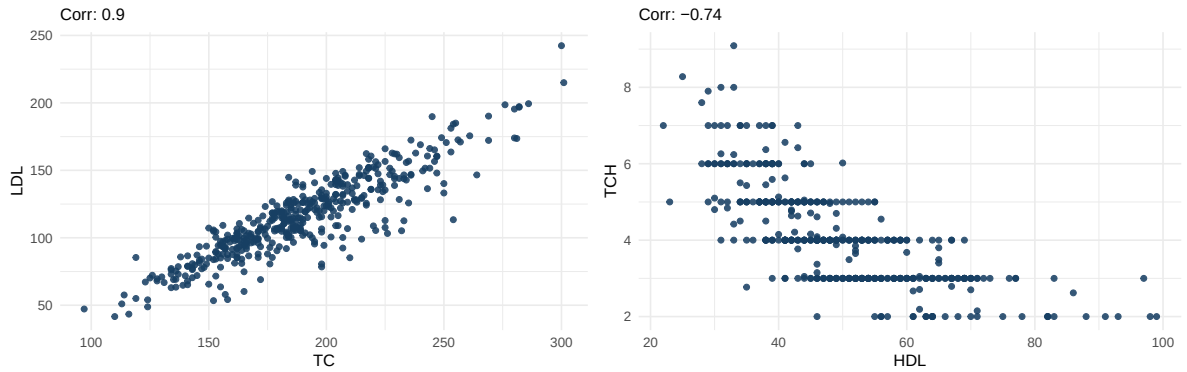


Figura 1

```
df_diabetes %>% ggplot(aes(x=tc, y=ldl)) +
  geom_point(color=color) +
  labs(x ="TC", y = "LDL",
       subtitle = paste0("Corr: ",
                         round(cor(df_diabetes$tc, df_diabetes$ldl), 2))) +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))


df_diabetes %>% ggplot(aes(x=hdl, y=tch)) +
```

```
  geom_point(color=color) +
  labs(x ="HDL", y = "TCH",
       subtitle = paste0("Corr: ",
                         round(cor(df_diabetes$hdl, df_diabetes$tch), 2))) +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
```



(a) TC vs LDL

(b) HDL vs TCH

Figura 2: Scatterplots

También noté que la variable SEX estaba en formato numérico, por lo que fue convertida a factor, además eliminé el Id.

```
df_diabetes_mf <- df_diabetes %>%
  select(-c("Id", "tc", "tch")) %>%
  mutate(sex = as.factor(sex))
```

Entonces, las variables a incluir en el modelo serían: AGE, SEX, BMI, BP, LDL, HDL, LTG y GLU. En la Figura 3 se ve que no parece haber diferencias entre las medianas de la variable dependiente para los distintos valores de SEX. Por otro lado, en la Figura 4 se ve la asociación entre las covariables y la progresión de la diabetes, donde se destacan BMI, LTG y BP como las covariables con la correlación más fuerte. Finalmente, en la Figura 5 vemos histogramas de todas las covariables numéricas.

```
df_diabetes_mf %>% ggplot(aes(y= y, x = sex)) +
  geom_boxplot(fill = color) +
  labs(title = "Boxplot de la diabetes por sexo",
       y = "y",
       x = "sex") +
  theme_minimal() +
```

```
  theme(plot.title = element_text(hjust = 0.5, size = 14),
        axis.title = element_text(size = 12),
        axis.text = element_text(size = 10),
        strip.text = element_text(size = 12))
```

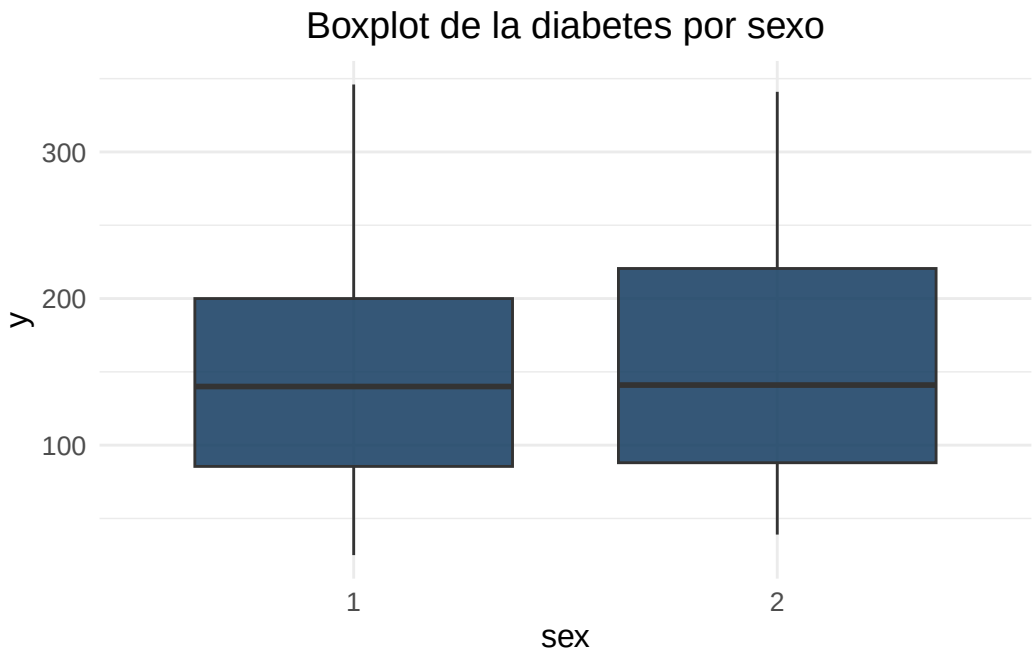
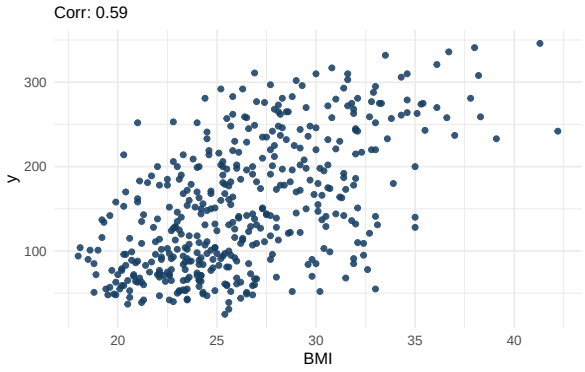
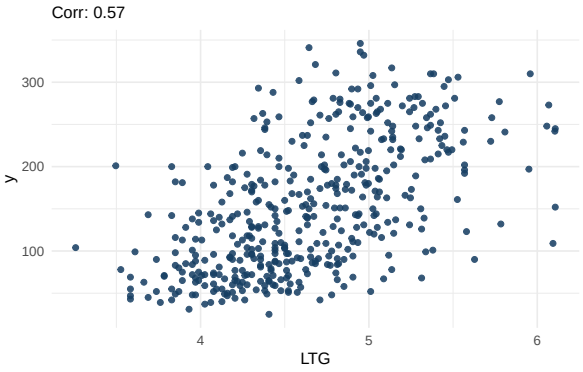


Figura 3: Boxplot de la edad para las dos categorías de la variable sex.

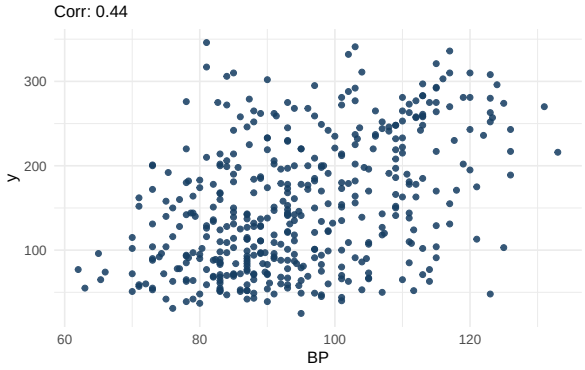
```
numeric_data_long <- df_diabetes_mf %>%
  select_if(is.numeric) %>%
  pivot_longer(cols = everything(), names_to = "variable",
               values_to = "value")

ggplot(numeric_data_long, aes(x = value)) +
  geom_histogram(bins = 20, fill = color, color = "white") +
  facet_wrap(~ variable, scales = "free_x") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
```



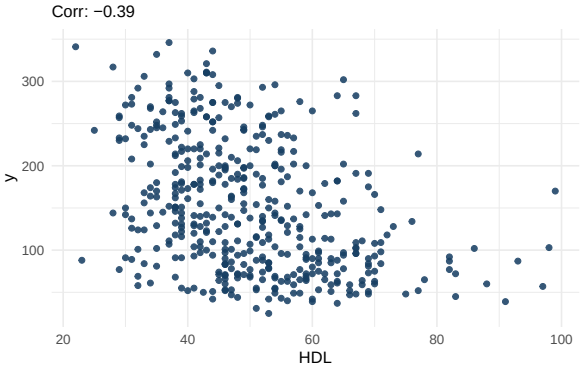
(a) bmi



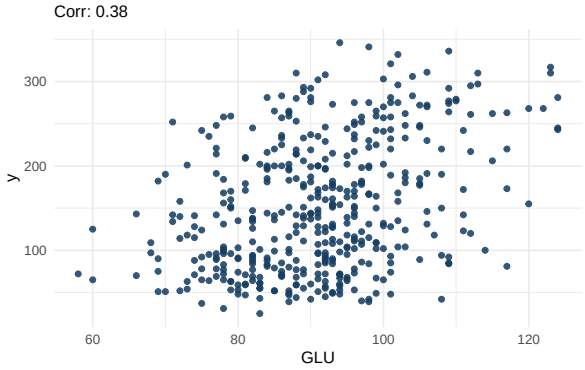
(b) ltg



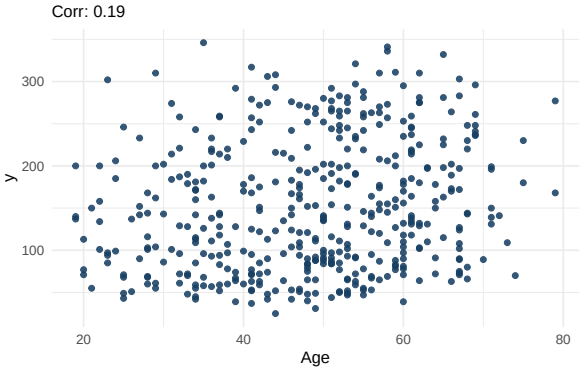
(c) bp



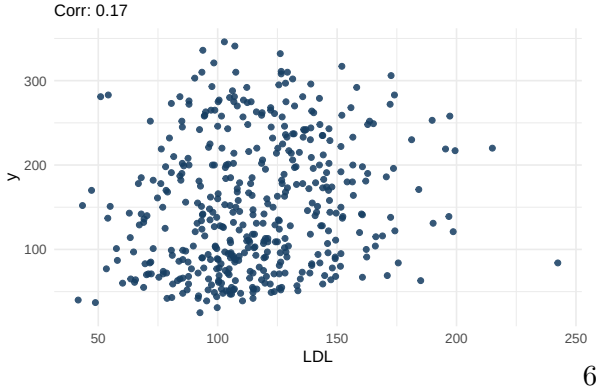
(d) hdl



(e) glu



(f) age



(g) ldl

Figura 4: Scatterplots de las covariables vs el índice de progresión de la diabetes.

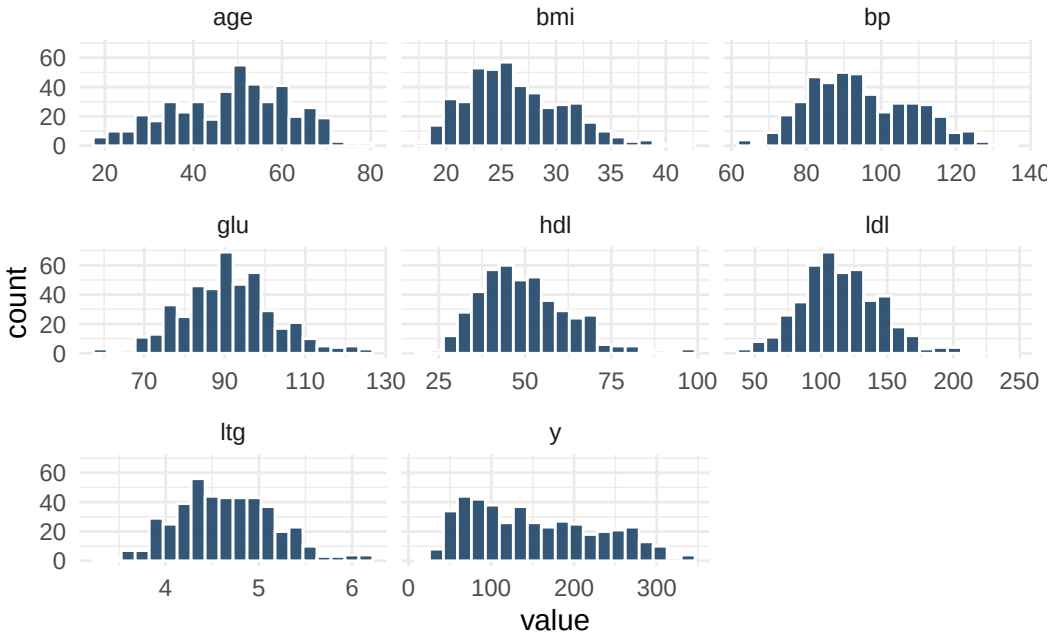


Figura 5: Scatterplots de las covariables vs el índice de progresión de la diabetes.

Antes de ajustar los modelos, dividí los datos entre train y test, asignando el 80% de los datos al conjunto de entrenamiento y el 20% restante al de testeo.

```
set.seed(seed)
split <- initial_split(df_diabetes_mf, prop = .8)

train_data <- training(split)

test_data <- testing(split)
```

Modelo lineal

Ajusté un modelo lineal para predecir el índice de progresión de la diabetes con todas las covariables descritas en la sección anterior. Para el ajuste se utilizaron los datos de entrenamiento y se ve en la tabla de resumen que todos los coeficientes son significativos a excepción de AGE y GLU.

```
lm_fit <- lm(y ~ ., data = train_data)

summary(lm_fit)$coefficients %>%
  round(2) %>%
  kable()
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-219.92	38.94	-5.65	0.00
age	-0.21	0.24	-0.89	0.37
sex2	-24.05	6.36	-3.78	0.00
bmi	5.81	0.77	7.55	0.00
bp	1.10	0.24	4.66	0.00
ldl	-0.24	0.10	-2.45	0.01
hdl	-1.14	0.25	-4.49	0.00
ltg	45.25	6.79	6.66	0.00
glu	0.13	0.30	0.41	0.68

A modo de diagnóstico calculé el VIF del ajuste y no parece haber problemas de colinealidad en las variables, dado que todos los valores son cercanos a 1.

```
vif_lm <- vif(lm_fit)
vif_lm
```

```
     age      sex      bmi       bp      ldl      hdl      ltg      glu
1.222571 1.277634 1.492356 1.421707 1.154288 1.479581 1.621432 1.460437
```

Ahora, para entender si las covariables que no resultaron siginficativas, lo son a nivel simulatáneo uso el comando ANOVA y veo que el p-valor del test F es 0.6478, por lo que debería descartar las variables del modelo.

```
lm_fit_2 <- lm(y ~ sex + bmi + bp + ldl + hdl + ltg, data = train_data)
anova(lm_fit_2, lm_fit)
```

```
Analysis of Variance Table

Model 1: y ~ sex + bmi + bp + ldl + hdl + ltg
Model 2: y ~ age + sex + bmi + bp + ldl + hdl + ltg + glu
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1    346 955275
2    344 952866  2    2408.2 0.4347 0.6478
```

Al descartar estas covariables se ve que todas las covariables son significativas al 5% y las que tienen un mayor efecto en la progesión de la diabetes parecen ser LTG, SEX y BMI.

```
summary(lm_fit_2)$coefficients %>%
  round(2) %>%
  kable()
```

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-215.41	37.45	-5.75	0.00
sex2	-24.62	6.25	-3.94	0.00
bmi	5.84	0.76	7.68	0.00
bp	1.07	0.23	4.69	0.00
ldl	-0.24	0.10	-2.48	0.01
hdl	-1.16	0.25	-4.58	0.00
ltg	45.27	6.49	6.97	0.00

Calculando el gráfico de residuos vs predichos (Figura 6) pareciera haber una cierta estructura en los residuos. Por lo que grafiqué los mismos vs. las covariables incluidas (Figura 7) y no incluidas (Figura 8) en el modelo[1], aunque no encontré ninguna estructura muy marcada.

```
resultados_modelo <- data.frame(
  "residuos" = lm_fit_2$residuals,
  "predichos" = lm_fit_2$fitted.values
)

resultados_modelo %>% ggplot(aes(x=predichos, y = residuos)) +
  geom_point(color = color)+
  theme_minimal() +
  labs(title = "Residuos vs predichos",
       x= "Valores predichos",
       y = "Residuos") +
```

[1]Me quedó pendiente hacer el mismo grafico versus las variables que excluí en el análisis exploratorio.

```
  theme(plot.title = element_text(hjust = 0.5)) +
  geom_hline(yintercept = 0, linetype = "dashed")
```

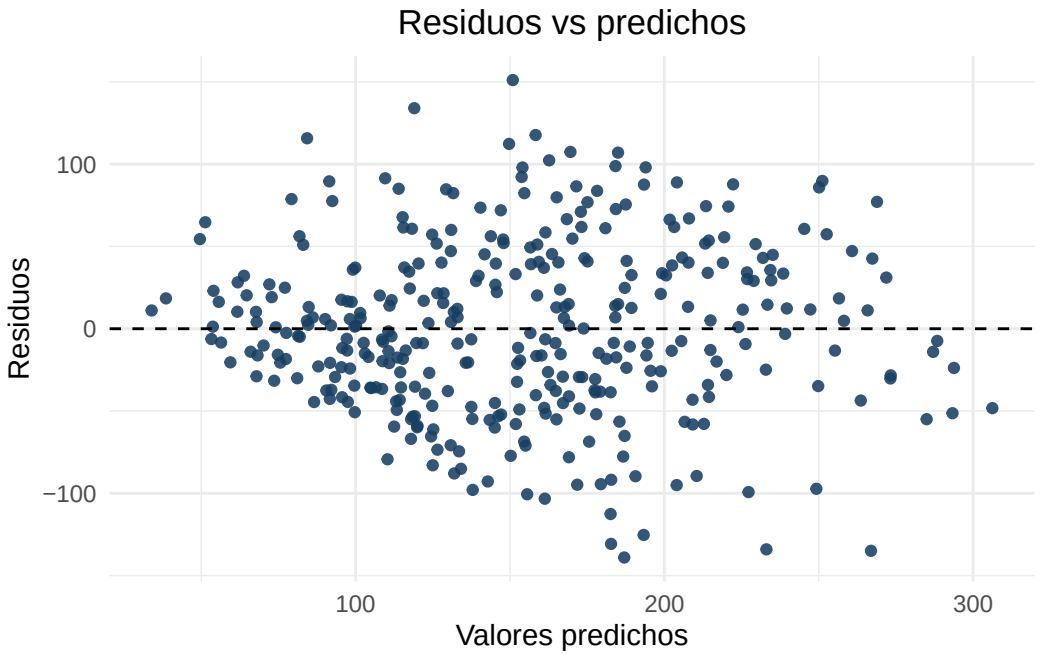


Figura 6: Grafico de residuos vs valores ajustados del modelo lineal.

```
plot(train_data$bmi, lm_fit_2$residuals, xlab= "BMI",
     ylab="Residuos", main="Residuos vs BMI", col = color)
plot(train_data$bp, lm_fit_2$residuals, xlab= "BP",
     ylab="Residuos", main="Residuos vs NP", col = color)
plot(train_data$ldl, lm_fit_2$residuals, xlab= "LDL",
     ylab="Residuos", main="Residuos vs LDL", col = color)
plot(train_data$hdl, lm_fit_2$residuals, xlab= "HDL",
     ylab="Residuos", main="Residuos vs HDL", col = color)
plot(train_data$ltg, lm_fit_2$residuals, xlab= "LTG",
     ylab="Residuos", main="Residuos vs LTG", col = color)
```

```
plot(train_data$glu, lm_fit_2$residuals, xlab= "GLU",
     ylab="Residuos", main="Residuos vs GLU", col = color)
plot(train_data$age, lm_fit_2$residuals, xlab= "Age",
     ylab="Residuos", main="Residuos vs Age", col = color)
```

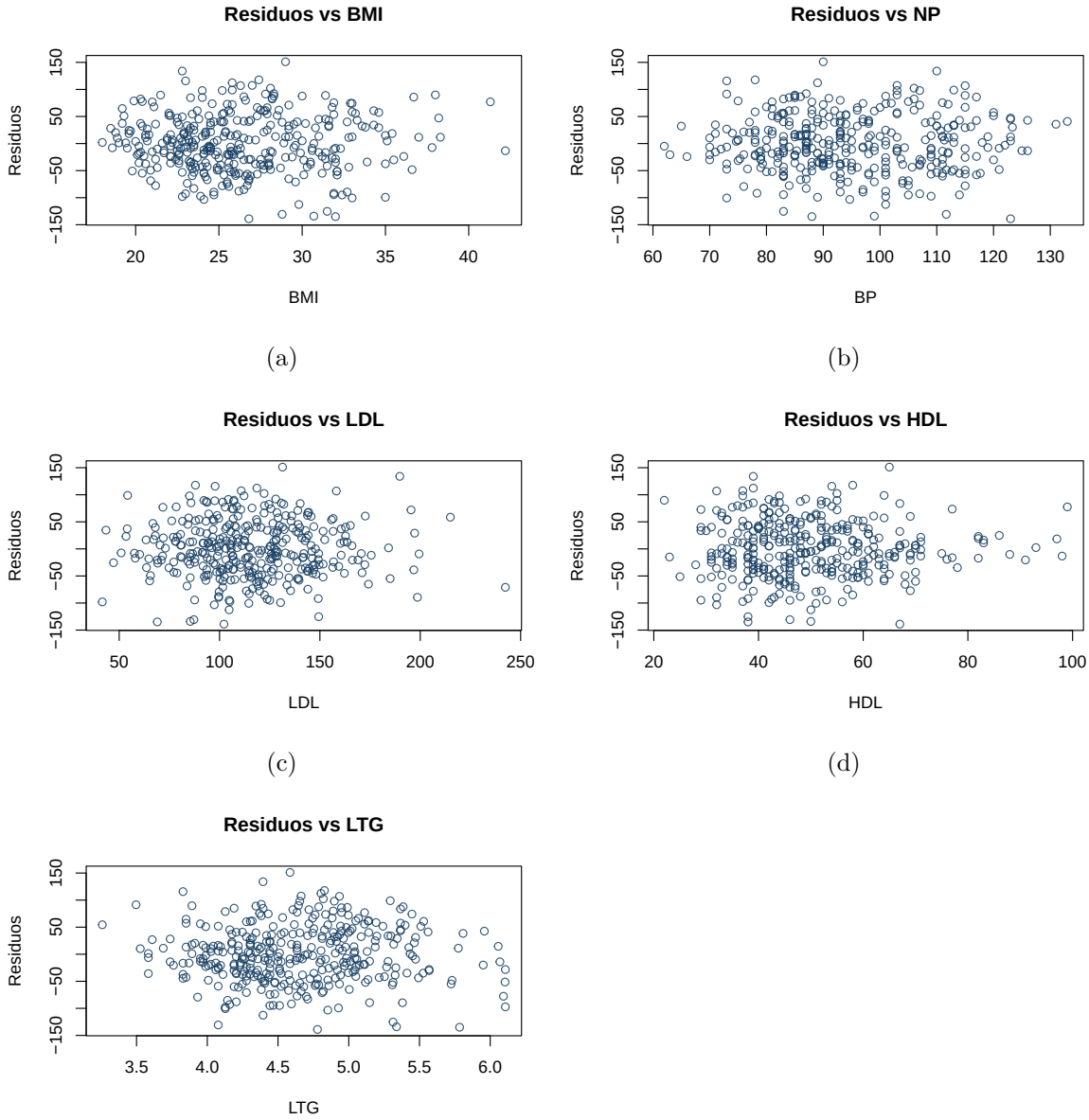
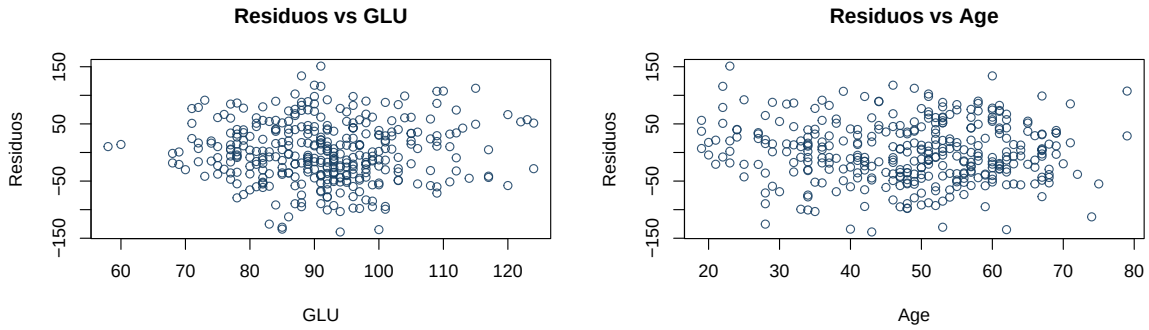


Figura 7: Graficos de residuos vs las covariables incluidas en el modelo



(a) (b)

Figura 8: Graficos de residuos vs las covariables excluidas del modelo

Finalmente, calculo el MSE en la muestra de testeo.

```
preds_lm <- predict(lm_fit_2, test_data)

mse_lm = mse(test_data$y, preds_lm)

cat(paste("MSE del modelo lineal:", round(mse_lm, 4)))
```

```
MSE del modelo lineal: 3685.7984
```

Random Forest

Ahora voy a ajustar un modelo Random Forest para comparar los resultados con el modelo lineal. En primer lugar ajusto con 1000 árboles $mtry = \sqrt{8}$ con la muestra de entrenamiento y calculo el MSE en la muestra de testeo.

```
set.seed(seed)

rf_model <- randomForest(y ~ ., data = train_data, ntree = 1000,
                         mtry = sqrt(ncol(train_data)-1))

preds_rf <- predict(rf_model, test_data)

mse_rf <- mse(test_data$y, preds_rf)

cat(paste("MSE de Random Forest:", round(mse_rf, 4)))
```

```
MSE de Random Forest: 3506.076
```

Se ve que el MSE es menor al obtenido en el modelo lineal.

Sin embargo, el objetivo no era únicamente predecir, sino también entender el efecto de cada covariable en la diabetes. Por lo que usé permutation feature importance y partial dependence plots para entender las importancias y los efectos de las covariables, respectivamente.

```
predictor <- Predictor$new(rf_model, data = train_data, y = "y")
```

Para lo primero se ve en la Figura 9 que las dos variables más importantes son LTG y BMI. A diferencia del modelo lineal, SEX tiene muy poca importancia, de todas formas esto parece ir en línea con el análisis exploratorio.

```
imp <- FeatureImp$new(predictor, loss = "mse")

imp$results %>%
  mutate_if(is.numeric, round, 4) %>%
  kable()
```

feature	importance.05	importance	importance.95	permutation.error
ltg	4.4112	4.6973	5.3425	2833.1680
bmi	4.2248	4.6080	4.9185	2779.3478

feature	importance.05	importance	importance.95	permutation.error
bp	2.2810	2.5047	2.5210	1510.7103
hdl	2.2529	2.4018	2.5528	1448.6703
ldl	1.7739	1.7971	1.8385	1083.9083
glu	1.7032	1.7750	1.8018	1070.5781
age	1.6338	1.6966	1.7483	1023.3252
sex	1.2771	1.3317	1.3515	803.2291

```
plot(imp) + theme_minimal()
```

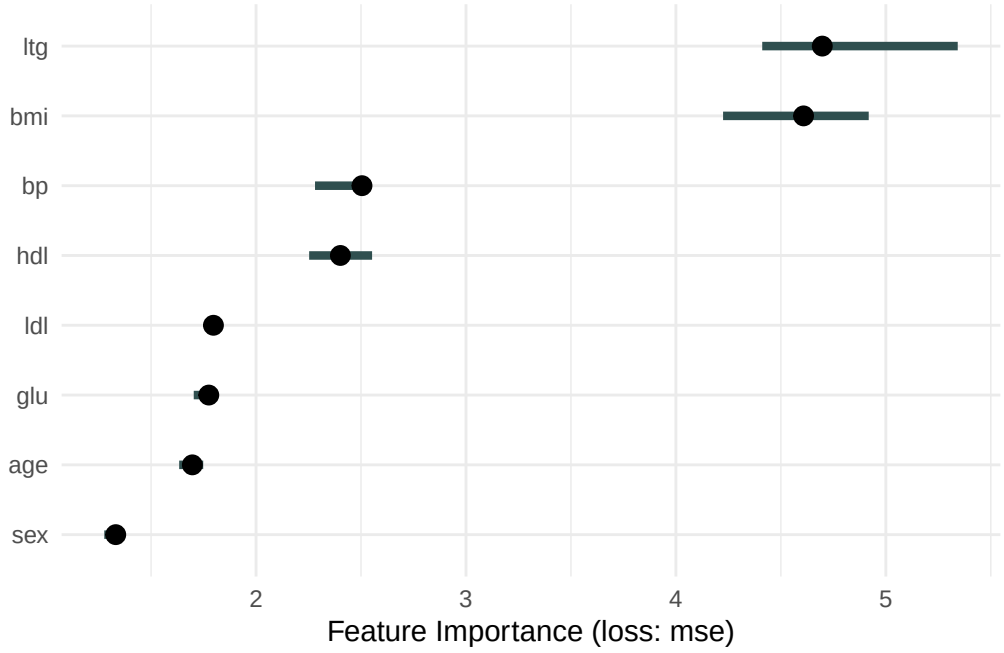
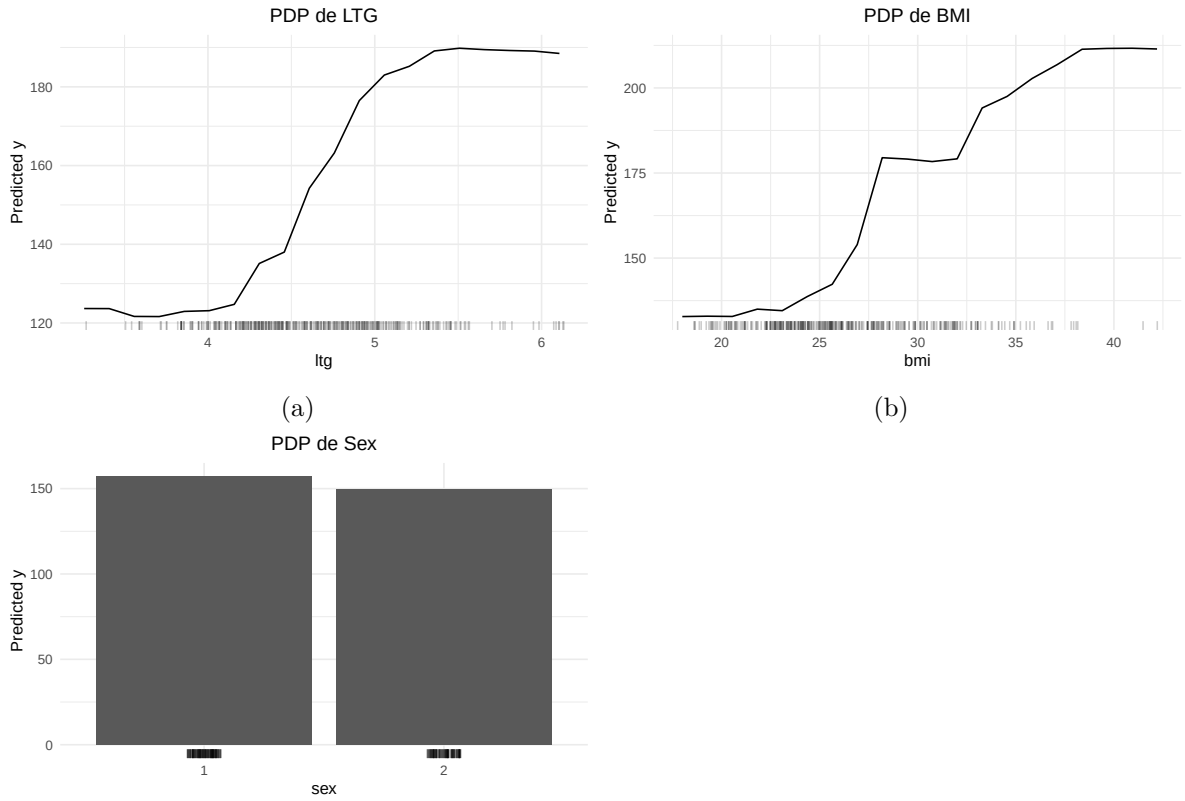


Figura 9: Importancias de Random Forest.

Para estudiar el efecto marginal de las variables podemos usar la Figura 10. Donde se ve que los efectos de las dos covariables más importantes (LTG y BMI) parecen ser positivos, lo cual iría en línea con los resultados obtenidos en el ajuste lineal. En la misma figura también se puede ver el gráfico de SEX que, en línea con lo obtenido en el modelo lineal, parece ser negativo.

```
pd_ltg <- FeatureEffect$new(predictor, feature = "ltg", method = "pdp")

pd_bmi <- FeatureEffect$new(predictor, feature = "bmi", method = "pdp")
```

```r
pd_sex <- FeatureEffect$new(predictor, feature = "sex", method = "pdp")

plot(pd_ltg) +
  theme_minimal() +
  labs(title="PDP de LTG") +
  theme(plot.title = element_text(hjust = 0.5))

plot(pd_bmi) +
  theme_minimal() +
  labs(title="PDP de BMI") +
  theme(plot.title = element_text(hjust = 0.5))

plot(pd_sex) +
  theme_minimal() +
  labs(title="PDP de Sex") +
  theme(plot.title = element_text(hjust = 0.5))
```



(a) (b)

(c)

Figura 10: Partial dependence plots de las dos covariables más importantes de Random Forest y SEX.

Conclusiones y pendientes

En este ejemplo busqué entender el efecto de distintas variables en la progresión de la diabates. Para ello, luego de un breve análisis exploratorio, ajusté un modelo lineal y un Random Forest para luego comparar sus resultados. Siendo este último el que tuvo un menor error en la muestra de testeo, aún sin hacer una búsqueda intensiva de hiperparámetros. En cuanto a la interpretación de los parámetros ambos modelos coinciden en que LTG y BMI son importantes, sin embargo SEX parece tener un efecto mayor en el ajuste lineal, mientras que en Random Forest fue la covariable menos importante.

Queda, entre muchas otras cosas, entender mejor qué causa la estructura en los residuos del modelo lineal y tratar de dar algún tipo de significancia estadística a los resultados de Random Forest. Además de hacer una busqueda de hiperámetros con validación cruzada y probar otras ténicas de interpretación como SHAP.

Referencias

Shmueli, G. (2010). To explain or to predict? Statistical Science, 25(3), 289-310.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning: with applications in R. Corrected edition. New York, Springer.

Hastie, T., Tibshirani, R., & Friedman, J. H. (2009). The elements of statistical learning: data mining, inference, and prediction. 2nd ed. New York, Springer.

Molnar, C. (2019). Interpretable Machine Learning.