Explainable survival analysis with **survxai** : : **CHEAT SHEET**

Intorduction

Survival analysis models are used primarily in medicine and churn analysis. Due to many applications, we are witnessing a fast development of a wide range of black-box survival models. Their lack of interpretability makes them unusable for analyzes that require an understanding of the model behavior.

An R package **survxai** is a tool for creating explanations of survival models. For both, complex and simple survival models. It also enables to compare them. Currently, four explanation methods are implemented. We can divide them into 2 groups: local and global.

Basics

Package survxai contains functions for creating a unified representation of a survival models. Such representations can be further processed by various survival explainers. Tools implemented in survxai help to understand how input variables are used in the model and what impact do they have on final model prediction.

We use the data set from *the Mayo Clinic trial in primary biliary cirrhosis (PBC) of the liver* originally from the **randomForestSRC [1]** package. The limited version of this data set, divided into test and train sets, is included in the **survxai** package. Let's fit a random forest for survival model and create an explainer.

```
library(survxai); library(randomForestSRC); library(survival)
data(pbcTrain); data(pbcTest)
set.seed(1994)

rf_model <- rfsrc(Surv(years, status)~., data = pbcTrain)
surve_rf <- explain(model = rf_model,
        label = "rfsrc",
        data = pbcTest[ ,-c(1, 5)],
         y = Surv(pbcTest$years, pbcTest$status))
```

REFERENCES

1. Ishwaran H., Kogalur U.B., Blackstone E.H. and Lauer M.S. (2008). *Random survival forests.* Ann. Appl. Statist. 2(3), 841--860.
2. Przemyslaw Biecek (2018). *ceterisParibus: Ceteris Paribus Profiles*. R package version 0.3.0. https://CRAN.R-project.org/package=ceterisParibus
3. Staniak M, Biecek P (2018). *Explanations of Model Predictions with live and breakDown Packages*. _ArXiv e-prints_. 1804.01955. https://arxiv.org/abs/1804.01955.
4. Brandon M. Greenwell (2017). *pdp: An R Package for Constructing Partial Dependence Plots.* The R Journal, 9(1), 421--436. URL https://journal.r-project.org/archive/2017/RJ-2017-016/index.html.
5. Mogensen, U. B., Ishwaran, H., & Gerds, T. A. (2012). *Evaluating Random Forests for Survival Analysis using Prediction Error Curves.* Journal of Statistical Software, 50(11), 1–23.

Local Explanations

CETERIS PARIBUS

Ceteris paribus plot [2] presents model responses around a single point in the feature space. Ceteris paribus for survival model consists of survival curves around one observation. Each curve represents a different value of the chosen variable. Ceteris paribus plot illustrates how may the survival curve change along with the changing values of the variable.

```
new_observation <- pbcTest[1, -c(1, 5)]
cp_rf <- ceteris_paribus(surve_rf, new_observation)
plot(cp_rf, selected_variable = "stage")
```
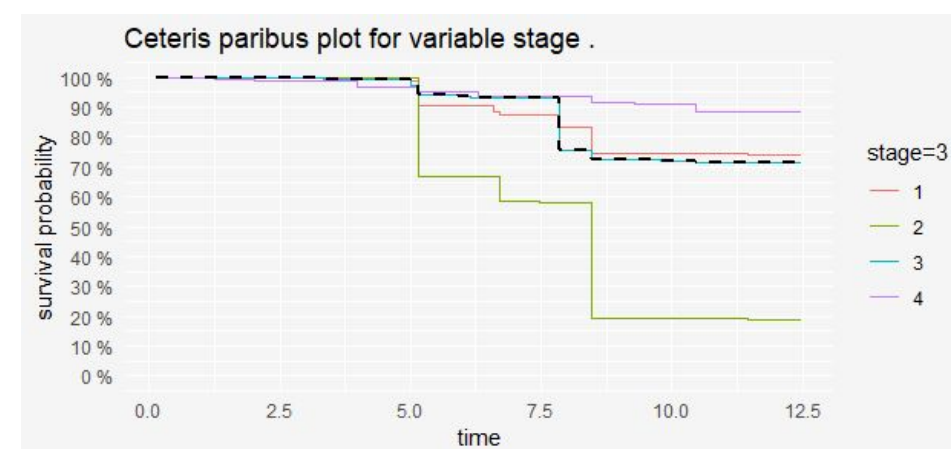

PREDICTION BREAKDOWN

Prediction breakdown plot [3] presents variable contributions in final predictions. Break down for survival model helps to understand which factors drive survival probability for a single observation.

```
new_observation <- pbcTest[1, -c(1, 5)]
pb_rf <- prediction_breakdown(surve_rf, new_observation)
plot(pb_rf)
```
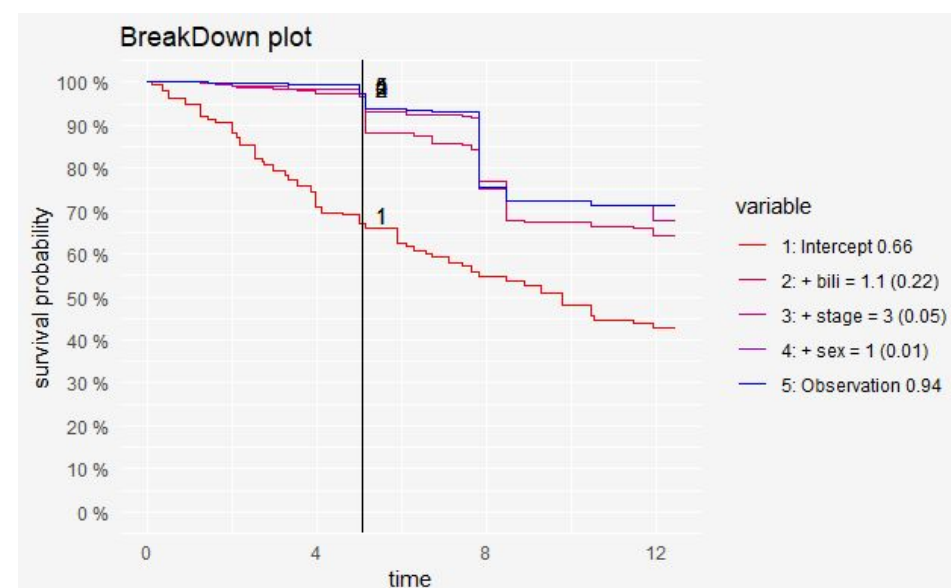

Global Explanations

VARIABLE RESPONSE

Variable response plot is designed to better understand the relation between a variable and a model output. Variable response plot illustrates how the mean survival curve change along with the changing values of the variable. It is inspired by **partial dependence plots [4]**.

```
vr_rf_sex <- variable_response(surve_rf, "sex")
plot(vr_rf_sex)
```
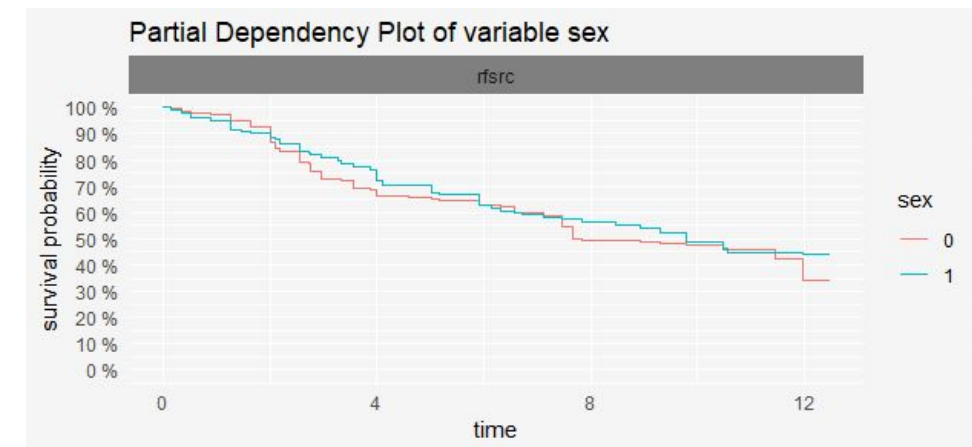

MODEL PERFORMANCE

Model performance curves present prediction error for the chosen survival model depending on time. For computing prediction error we use the expected **Brier Score [5]**.

```
library(rms)
cph_model <- cph(Surv(years, status)~., data = pbcTrain,
surv = TRUE, x = TRUE, y=TRUE)
surve_cph <- explain(model = cph_model,
        data = pbcTest[ ,-c(1, 5)],
         y = Surv(pbcTest$years, pbcTest$status))

mp_cph <- model_performance(surve_cph)
mp_rf <- model_performance(surve_rf)
plot(mp_cph, mp_rf) + ylim(c(0, 0.30))
```