# EDS230: Assignment 4

Carlo Broderick, Erika Egg, Ruth Enriquez

```r
# Load packages
library(sensitivity)
library(tidyverse)
library(lhs)
library(purrr)
```

**Assignment Tasks**
For a given forest, you will perform a sensitivity analysis of model predictions of conductance. Consider the sensitivity of your estimate to uncertainty in the following parameters and inputs

- height
- kd
- k0
- v

Windspeeds v are normally distributed with a mean of 250 cm/s with a standard deviation of 30 cm/s
For vegetation height assume that height is somewhere between 9.5 and 10.5 m (but any value in that range is equally likely)
For the kd and k0 parameters you can assume that they are normally distributed with standard deviation of 1% of their default values

```r
# Source the function
source("Catm-1.R")
```

a) Use the Latin hypercube approach to generate parameter values for the 4 parameters

```r
# Set a random seed to make things 'random' reproducibly
set.seed(2)

# Specify parameters
pnames = c("v", "height", "k_o", "k_d")

# Gather how many parameters
npar =  length(pnames)

# Choose how many samples
nsample = 50

# Create the random values array matrix using LHS for the parameters
parm_quant = randomLHS(nsample, npar)
```

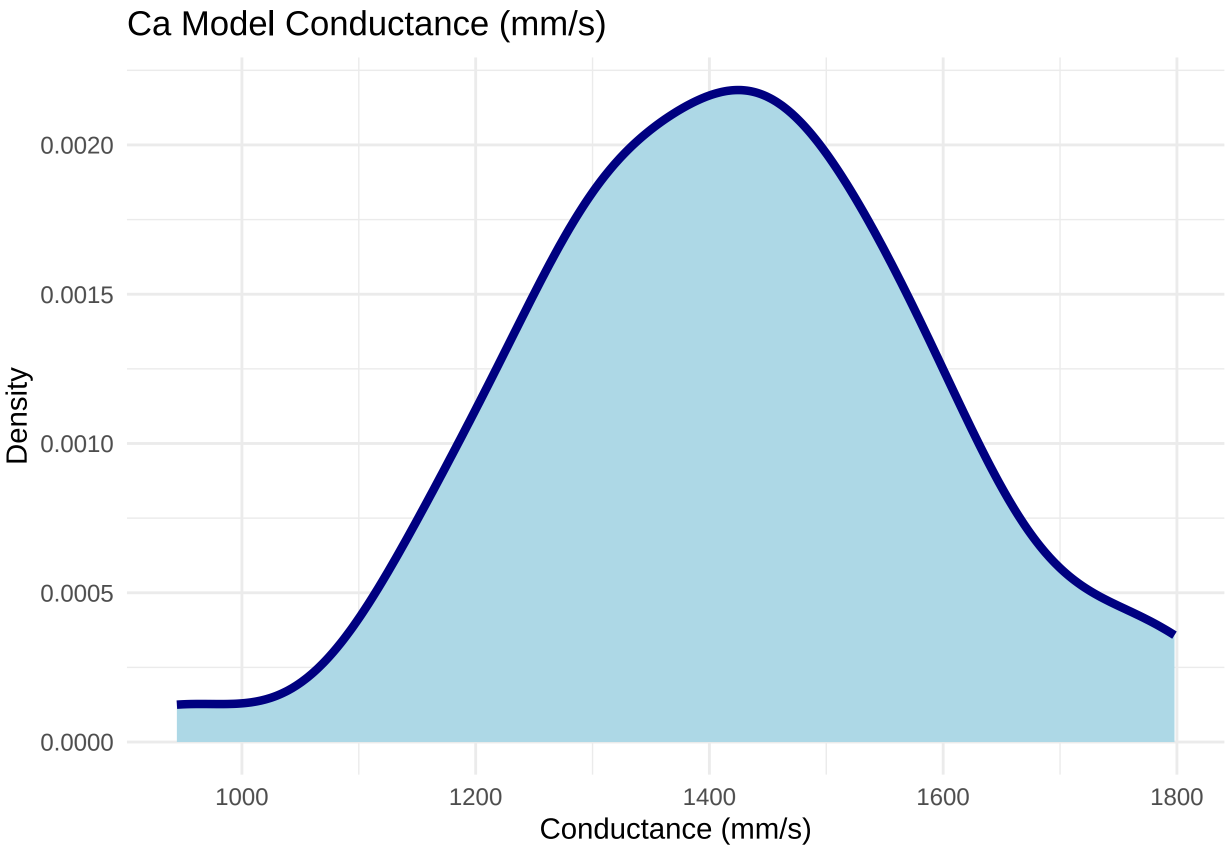
```r
# Assign the parameter names columns
colnames(parm_quant) = pnames

# Set up a data frame
parm = as.data.frame(matrix(nrow = nrow(parm_quant),
                            ncol = ncol(parm_quant)))

# Name data frame columns
colnames(parm) = pnames

# Create the samples for the different parameters

# To create the 1% standard deviation in next step
pvar = 100

# Normally distributed param samples
parm[,"v"] = qnorm(parm_quant[,"v"],
                   mean = 250,
                   sd = 30)
parm[,"k_d"] = qnorm(parm_quant[,"k_d"],
                     mean = 0.7,
                     sd = 0.7/pvar)
parm[,"k_o"] = qnorm(parm_quant[,"k_o"],
                     mean = 0.1,
                     sd = 0.1/pvar)

# Uniformly distributed param samples
parm[,"height"] = qunif(parm_quant[,"height"],
                        min = 9.5,
                        max = 10.5)
```

b) Run the atmospheric conductance model for these parameters

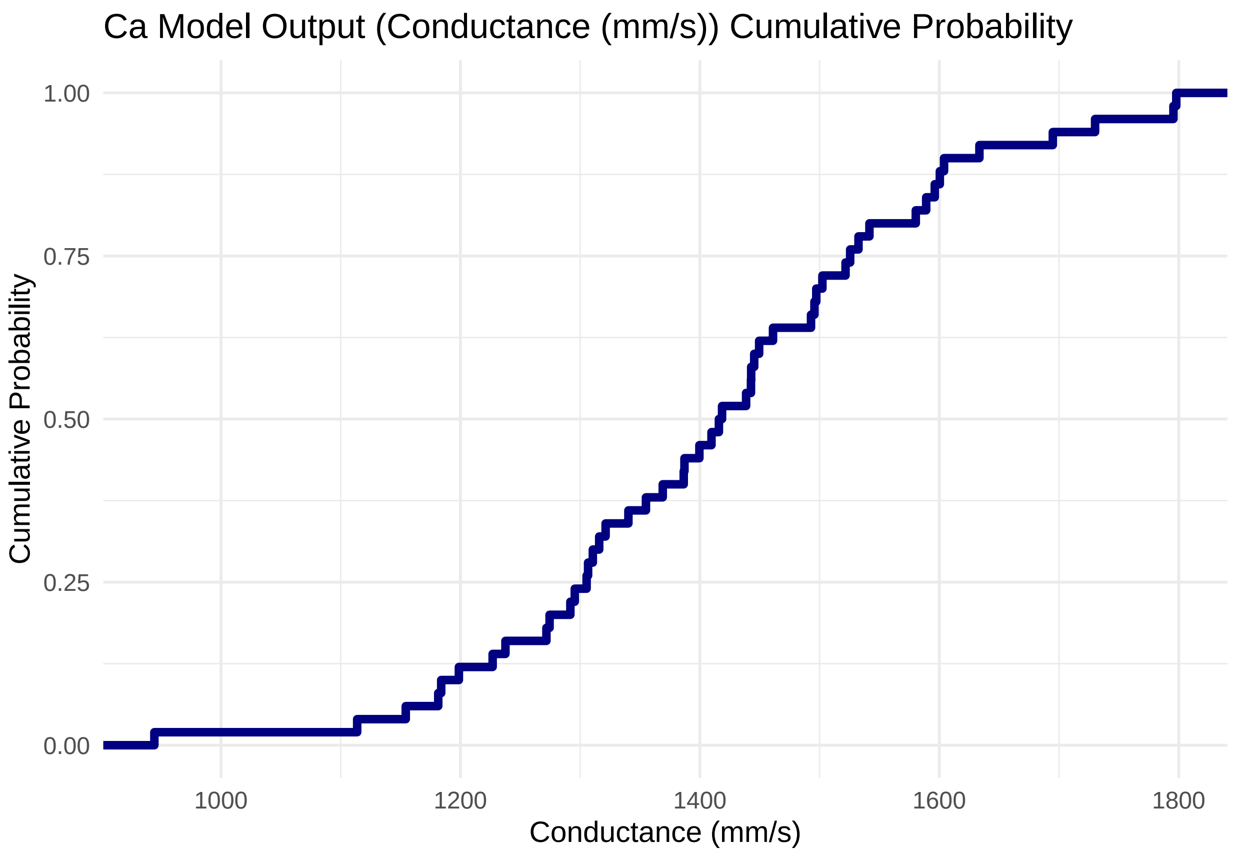
```r
# Run the hypercube through the model
Ca_outputs = pmap(parm, Catm)

# Turn results into an array for easy display/analysis
Cas = unlist(Ca_outputs)

# Put the outputs in the same df as the parameters
param_outputs <- parm %>%
  mutate(output = Cas)
```

c) Plot conductance estimates in a way that accounts for parameter uncertainty

```r
# Plot predicted Ca for each parameter combo
ggplot(param_outputs, aes(x = output)) +
  geom_density(color = "navy",
               size = 1.5,
               fill = "lightblue") +
  theme_minimal() +
  labs(x = "Conductance (mm/s)",
       y = "Density",
       title = "Ca Model Conductance (mm/s)")
```

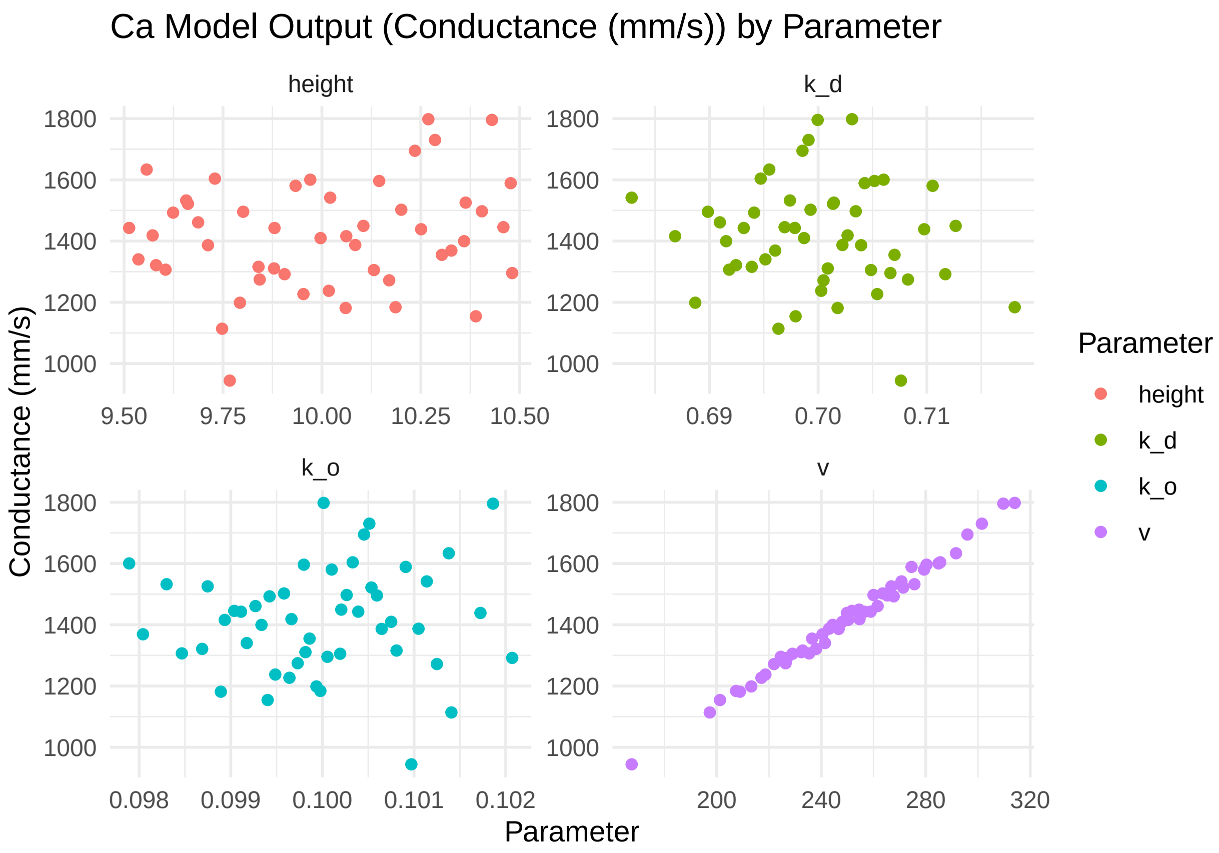


```
# Plot cumulative distribution
ggplot(param_outputs, aes(x = output)) +
  stat_ecdf(color = "navy",
            size = 1.5) +
  theme_minimal() +
  labs(x = "Conductance (mm/s)",
       y = "Cumulative Probability",
       title = "Ca Model Output (Conductance (mm/s)) Cumulative Probability")
```



d) Plot conductance estimates against each of your parameters

```
# Make a df for the outputs pivoted longer for graphs
df_long <- param_outputs %>%
  pivot_longer(cols = v:k_d, names_to = "parm", values_to = "value")

# Create plots for parameters effect on output
ggplot(df_long, aes(x = value, y = output, col = parm)) +
  geom_point(size = 1.5) +
  facet_wrap(~parm,
             ncol = 2,
             scales = "free") +
  theme_minimal() +
  labs(x = "Parameter",
       y = "Conductance (mm/s)",
       color = "Parameter",
       title = "Ca Model Output (Conductance (mm/s)) by Parameter")
```

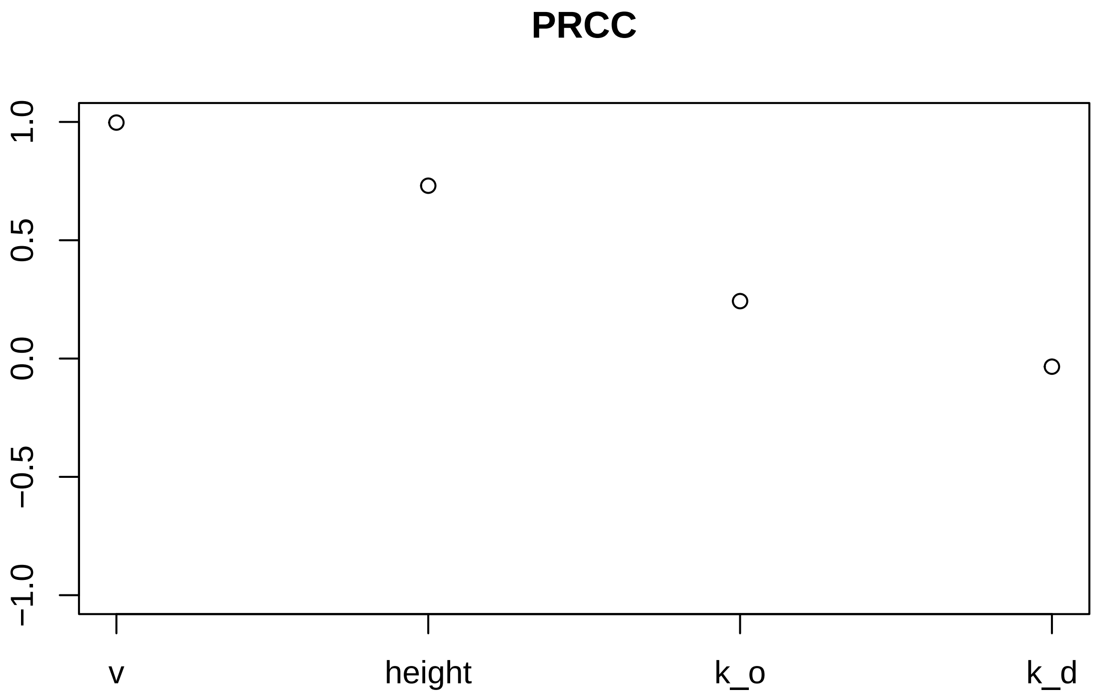


e) Estimate the Partial Rank Correlation Coefficients

```
# Calculate partial correlations
partial_correlation = pcc(parm, param_outputs$output, rank = TRUE)

# Display values
partial_correlation
```

```
##
## Call:
## pcc(X = parm, y = param_outputs$output, rank = TRUE)
##
## Partial Rank Correlation Coefficients (PRCC):
##           original
## v        0.9973597
## height   0.7304391
## k_o      0.2426520
## k_d     -0.0342727
```

```
# Plot them
plot(partial_correlation)
```



f) Discuss what your results tell you about how aerodynamic conductance? What does it suggest about what you should focus on if you want to reduce uncertainty in aerodynamic conductance estimates? Does this tell you anything about the sensitivity of plant water use to climate change?

**Answer: We suggest to reduce uncertainty that we look at wind speed and height, as they are the two most important values and are most sensitive to change. This can be seen in the partial correlation graph which shows them as the highest values and in the obvious positive correlations seen in the output by parameter plot. Since wind speed and height seem to be most sensitive to change, uncertainty in these parameters may be more likely to create an underestimate or overestimate of plant water use under climate change.**