Elementary Effects for the BTD Model

Setup packages.

In [1]:
```r
require(data.table)
require(magrittr)
require(sensitivity)

require(ggplot2)
```

```
Loading required package: data.table
Loading required package: magrittr
Loading required package: sensitivity
Registered S3 method overwritten by 'sensitivity':
  method    from
  print.src dplyr
Loading required package: ggplot2
```

Design experiment.

Load input ranges.

In [ ]:
```r
z.ranges <- fread("input-ranges.tsv")
z.ranges %>% dim
```

In [ ]:
```r
z.ranges %>% summary
```

One-at a time experiment with 500 repetitions, à la Morris.

```
In [ ]: z.design <- morris(
    NULL,
    factors = z.ranges$Variable,
    r = 500,
    design = list(
        type = "oat",
        levels = mapply(function(t, x0, x1) {
            if (t == "Integer")
                x1 - x0 + 1
            else if (t == "Boolean")
                2
            else
                5
        }, z.ranges$Type, z.ranges$`Sensitivity Minimum`, z.ranges$`Sens
itivity Maximum`),
        grid.jump = 1
    )
)
z.design$X %>% dim
```

```
In [ ]: write.table(z.design$X, file = "design.tsv", row.names = FALSE, col.name
s = TRUE, sep = "\t", quote = FALSE)
```

Relate the design to the model's variables.

```
In [ ]: z.inputs <- cbind(
    Run = 1:(dim(z.design$X)[1]),
    data.table(
        sweep(
            sweep(z.design$X, MARGIN = 2, z.ranges$`Sensitivity Maximum`
- z.ranges$`Sensitivity Minimum`, `*`),
            MARGIN = 2,
            z.ranges$`Sensitivity Minimum`,
            `+`
        )
    )
)
z.inputs %>% summary
```

```
In [ ]: write.table(z.inputs, file="inputs.tsv", row.names = FALSE, col.names =
TRUE, sep = "\t", quote = FALSE)
```

Analyze results.

Read files.

Read design.

In [2]:
```
z.design <- fread("design.tsv")
z.design %>% dim
```

42500 84

Read inputs.

In [3]:
```
z.inputs <- fread("inputs.tsv")
z.inputs %>% dim
```

42500 85

Read outputs.

In [4]:
```
z.outputs <- fread("outputs.tsv")
z.outputs[Time == 2050] %>% dim
```

42500 51

In [5]:
```
z.outputs[Time == 2050] %>% summary
```

```
      Run             Time      bioproduct market share mass
 Min.   :    1   Min.   :2050   Min.   :-27.65680
 1st Qu.:10626   1st Qu.:2050   1st Qu.:  0.01149
 Median :21250   Median :2050   Median :  0.33107
 Mean   :21250   Mean   :2050   Mean   :  0.59533
 3rd Qu.:31875   3rd Qu.:2050   3rd Qu.:  0.71766
 Max.   :42500   Max.   :2050   Max.   : 54.99460
 current market size economic current market size mass long term market
share
 Min.   :-9.881e+13          Min.   :1.750e+04        Min.   :0.0000
 1st Qu.:-2.500e+10          1st Qu.:2.069e+07        1st Qu.:0.2126
 Median : 7.524e+10          Median :6.609e+07        Median :0.3172
 Mean   : 3.949e+11          Mean   :2.866e+08        Mean   :0.3959
 3rd Qu.: 5.717e+11          3rd Qu.:2.009e+08        3rd Qu.:0.5406
 Max.   : 5.844e+13          Max.   :7.965e+09        Max.   :0.9810
 long term market value    Adopters            NonAdopters
 Min.   :0.000e+00      Min.   : -11334900   Min.   :5.130e+02
 1st Qu.:0.000e+00      1st Qu.:    416403   1st Qu.:5.613e+06
 Median :0.000e+00      Median :   7035080   Median :2.763e+07
 Mean   :2.098e+09      Mean   : 135755526   Mean   :1.205e+08
 3rd Qu.:7.470e+02      3rd Qu.:  59639400   3rd Qu.:9.357e+07
 Max.   :6.297e+11      Max.   :5976570000   Max.   :5.388e+09
 Potential Adopters   abandoning bioproduct Cumulative Demoing Producti
on
 Min.   : -20212000   Min.   :0             Min.   :       0
 1st Qu.:     20273   1st Qu.:0             1st Qu.:       0
 Median :   1188780   Median :0             Median :       0
 Mean   :  30356549   Mean   :0             Mean   :  182732
 3rd Qu.:  19697400   3rd Qu.:0             3rd Qu.:       0
 Max.   :2077400000   Max.   :0             Max.   :11109600
 Cumulative Production  prepiloting     pilot plant construction
 Min.   :       0      Min.   :0.0000   Min.   :0.0000000
 1st Qu.:       0      1st Qu.:0.0000   1st Qu.:0.0000000
 Median :       0      Median :0.0000   Median :0.0000000
 Mean   :  696871      Mean   :0.2168   Mean   :0.0002118
 3rd Qu.:       0      3rd Qu.:0.0000   3rd Qu.:0.0000000
 Max.   :67509600      Max.   :1.0000   Max.   :1.0000000
 pilot plant is built startup piloting complete piloting ongoing
 Min.   :0.000        Min.   :0.0000            Min.   :0.0000
 1st Qu.:1.000        1st Qu.:1.0000            1st Qu.:0.0000
 Median :1.000        Median :1.0000            Median :1.0000
 Mean   :0.783        Mean   :0.7767            Mean   :0.5731
 3rd Qu.:1.000        3rd Qu.:1.0000            3rd Qu.:1.0000
 Max.   :1.000        Max.   :1.0000            Max.   :1.0000
 piloting progress piloting complete   predemoing      demo plant const
ruction
 Min.   :0.00      Min.   :0.0000    Min.   :0.00000   Min.   :0.000000
 1st Qu.:0.00      1st Qu.:0.0000    1st Qu.:0.00000   1st Qu.:0.000000
 Median :0.00      Median :0.0000    Median :0.00000   Median :0.000000
 Mean   :0.21      Mean   :0.2099    Mean   :0.06772   Mean   :0.003106
 3rd Qu.:0.00      3rd Qu.:0.0000    3rd Qu.:0.00000   3rd Qu.:0.000000
 Max.   :1.00      Max.   :1.0000    Max.   :1.00000   Max.   :1.000000
 demo plant is built regulatory process ongoing startup demoing complet
ed
 Min.   :0.0000      Min.   :0.00               Min.   :0.0000
 1st Qu.:0.0000      1st Qu.:0.00               1st Qu.:0.0000
 Median :0.0000      Median :0.00               Median :0.0000
```

```
 Mean   :0.1372      Mean   :0.01              Mean   :0.1329  
 3rd Qu.:0.0000      3rd Qu.:0.00              3rd Qu.:0.0000  
 Max.   :1.0000      Max.   :1.00              Max.   :1.0000  
 demoing ongoing   demoing progress  demoing complete  regulatory delay
 Min.   :0.00000   Min.   :0.00000   Min.   :0.00000   Min.   :0.0000  
 1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.00000   1st Qu.:0.0000  
 Median :0.00000   Median :0.00000   Median :0.00000   Median :0.0000  
 Mean   :0.09219   Mean   :0.05074   Mean   :0.04504   Mean   :0.0024  
 3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:0.00000   3rd Qu.:0.0000  
 Max.   :1.00000   Max.   :1.00000   Max.   :1.00000   Max.   :1.0000  
 precommercial      commercial plant construction commercial plant is b
uilt
 Min.   :0.000000   Min.   :0.000000              Min.   :0.00000   
 1st Qu.:0.000000   1st Qu.:0.000000              1st Qu.:0.00000   
 Median :0.000000   Median :0.000000              Median :0.00000   
 Mean   :0.007906   Mean   :0.003294              Mean   :0.03144   
 3rd Qu.:0.000000   3rd Qu.:0.000000              3rd Qu.:0.00000   
 Max.   :1.000000   Max.   :1.000000              Max.   :1.00000   
 commercial plant operation technology readiness level stage in progres
s
 Min.   :0.00000           Min.   :6.000             Min.   : 2.000  
 1st Qu.:0.00000           1st Qu.:6.000             1st Qu.: 5.000  
 Median :0.00000           Median :6.000             Median : 6.000  
 Mean   :0.03071           Mean   :6.311             Mean   : 8.823  
 3rd Qu.:0.00000           3rd Qu.:6.000             3rd Qu.:10.000  
 Max.   :1.00000           Max.   :9.000             Max.   :58.000  
   BS equity          payback period     NPV at required return
 Min.   :-2.075e+09   Min.   :    0.00   Min.   :-3.664e+10    
 1st Qu.:-1.073e+07   1st Qu.:    8.78   1st Qu.:-8.530e+08    
 Median :-1.996e+06   Median :   12.02   Median :-2.630e+08    
 Mean   : 2.478e+09   Mean   :   28.52   Mean   : 8.902e+08    
 3rd Qu.: 1.777e+08   3rd Qu.:   15.89   3rd Qu.: 8.030e+06    
 Max.   : 4.108e+11   Max.   :34389.80   Max.   : 2.937e+11    
 profitability indicator bioproduct favorability indicator
 Min.   :0.0000          Min.   :0.0000                   
 1st Qu.:1.0000          1st Qu.:0.0000                   
 Median :1.0000          Median :0.0000                   
 Mean   :0.9214          Mean   :0.2349                   
 3rd Qu.:1.0000          3rd Qu.:0.0000                   
 Max.   :1.0000          Max.   :1.0000                   
 long term selling price without green premium after market entry
 Min.   : 369.7                                                  
 1st Qu.:1399.2                                                  
 Median :2078.5                                                  
 Mean   :2136.7                                                  
 3rd Qu.:2768.2                                                  
 Max.   :6965.5                                                  
 total approval cost total approval time in business indicator
 Min.   :-1599210    Min.   : 0.019      Min.   :0.0000       
 1st Qu.:  199833    1st Qu.: 1.301      1st Qu.:0.0000       
 Median : 1052730    Median : 3.086      Median :0.0000       
 Mean   : 1636119    Mean   : 3.762      Mean   :0.4059       
 3rd Qu.: 2626350    3rd Qu.: 5.521      3rd Qu.:1.0000       
 Max.   :12420100    Max.   :20.166      Max.   :1.0000       
 internal project cancelled indicator   investing            granting    
 Min.   :0.0000000                    Min.   :0.000e+00   Min.   :0.000
e+00  
```

```
 1st Qu.:0.0000000                      1st Qu.:0.000e+00   1st Qu.:0.000
e+00
 Median :0.0000000                      Median :0.000e+00   Median :0.000
e+00
 Mean   :0.0006588                      Mean   :2.162e+08   Mean   :5.044
e+08
 3rd Qu.:0.0000000                      3rd Qu.:0.000e+00   3rd Qu.:3.192
e+07
 Max.   :1.0000000                      Max.   :2.189e+11   Max.   :9.984
e+10
 Total Government Grants Total Investment    Working Capital
 Min.   :0.000e+00       Min.   :3.000e+06   Min.   :-2.075e+09
 1st Qu.:0.000e+00       1st Qu.:1.145e+08   1st Qu.:-1.073e+07
 Median :1.373e+08       Median :7.370e+08   Median :-1.996e+06
 Mean   :6.579e+08       Mean   :1.274e+09   Mean   : 2.478e+09
 3rd Qu.:3.661e+08       3rd Qu.:1.488e+09   3rd Qu.: 1.777e+08
 Max.   :3.754e+10       Max.   :4.533e+10   Max.   : 4.108e+11
 IS production incentive
 Min.   :0.000e+00
 1st Qu.:0.000e+00
 Median :0.000e+00
 Mean   :1.075e+08
 3rd Qu.:0.000e+00
 Max.   :1.261e+10
```

Compute elementary effects.

Just use the final year.

In [6]:
```
z.outputs.clean <- z.outputs[order(Run)][`Time` == 2050, c(-1, -2)] %>%
as.matrix
z.outputs.clean %>% dim
```

42500 49

Define functions to compute elementary effects.

In [7]:

```
ind.rep <- function(i, p) {
# indices of the points of the ith trajectory in the DoE
  (1 : (p + 1)) + (i - 1) * (p + 1)
}

ee.oat <- function(X, y) {
  # compute the elementary effects for a OAT design
  p <- ncol(X)
  r <- nrow(X) / (p + 1)

#  if(is(y,"numeric")){
  if(inherits(y, "numeric")){
    one_i_vector <- function(i){
      j <- ind.rep(i, p)
      j1 <- j[1 : p]
      j2 <- j[2 : (p + 1)]
      # return((y[j2] - y[j1]) / rowSums(X[j2,] - X[j1,]))
      return(solve(X[j2,] - X[j1,], y[j2] - y[j1]))
    }
    ee <- vapply(1:r, one_i_vector, FUN.VALUE = numeric(p))
    ee <- t(ee)
    # "ee" is now a (r times p)-matrix.
#  } else if(is(y,"matrix")){
  } else if(inherits(y, "matrix")){
    one_i_matrix <- function(i){
      j <- ind.rep(i, p)
      j1 <- j[1 : p]
      j2 <- j[2 : (p + 1)]
      return(solve(X[j2,] - X[j1,],
                   y[j2, , drop = FALSE] - y[j1, , drop = FALSE]))
    }
    ee <- vapply(1:r, one_i_matrix,
                 FUN.VALUE = matrix(0, nrow = p, ncol = dim(y)[2]))
    # Special case handling for p == 1 and ncol(y) == 1 (in this case,
 "ee" is
    # a vector of length "r"):
    if(p == 1 && dim(y)[2] == 1){
      ee <- array(ee, dim = c(r, 1, 1))
    }
    # Transpose "ee" (an array of dimensions c(p, ncol(y), r)) to an arr
ay of
    # dimensions c(r, p, ncol(y)) (for better consistency with the stand
ard
    # case that "class(y) == "numeric""):
    ee <- aperm(ee, perm = c(3, 1, 2))
#  } else if(is(y,"array")){
  } else if(inherits(y, "array")){
    one_i_array <- function(i){
      j <- ind.rep(i, p)
      j1 <- j[1 : p]
      j2 <- j[2 : (p + 1)]
      ee_per_3rd_dim <- sapply(1:(dim(y)[3]), function(idx_3rd_dim){
        y_j2_matrix <- y[j2, , idx_3rd_dim]
        y_j1_matrix <- y[j1, , idx_3rd_dim]
        # Here, the result of "solve(...)" is a (p times dim(y)[2])-matr
ix or
```

```
        # a vector of length dim(y)[2] (if p == 1):
        solve(X[j2,] - X[j1,], y_j2_matrix - y_j1_matrix)
      }, simplify = "array")
      if(dim(y)[2] == 1){
        # Correction needed if dim(y)[2] == 1, so "y_j2_matrix" and
        # "y_j1_matrix" have been dropped to matrices (or even vectors,
 if also
        # p == 1):
        ee_per_3rd_dim <- array(ee_per_3rd_dim,
                                dim = c(p, dim(y)[2], dim(y)[3]))
      } else if(p == 1){
        # Correction needed if p == 1 (and dim(y)[2] > 1), so "y_j2_matr
ix" and
        # "y_j1_matrix" have been dropped to matrices:
        ee_per_3rd_dim <- array(ee_per_3rd_dim,
                                dim = c(1, dim(y)[2], dim(y)[3]))
      }
      # "ee_per_3rd_dim" is now an array of dimensions
      # c(p, dim(y)[2], dim(y)[3]). Assign the corresponding names for t
he
      # third dimension:
      if(is.null(dimnames(ee_per_3rd_dim))){
        dimnames(ee_per_3rd_dim) <- dimnames(y)
      } else{
        dimnames(ee_per_3rd_dim)[[3]] <- dimnames(y)[[3]]
      }
      return(ee_per_3rd_dim)
    }
    ee <- sapply(1:r, one_i_array, simplify = "array")
    # Special case handling if "ee" has been dropped to a vector:
#    if(is(ee,"numeric")){
    if (inherits(ee, "numeric")){
      ee <- array(ee, dim = c(p, dim(y)[2], dim(y)[3], r))
      dimnames(ee) <- list(NULL, dimnames(y)[[2]], dimnames(y)[[3]], NUL
L)
    }
    # "ee" is an array of dimensions c(p, dim(y)[2], dim(y)[3], r), so i
t is
    # transposed to an array of dimensions c(r, p, dim(y)[2], dim(y)
[3]):
    ee <- aperm(ee, perm = c(4, 1, 2, 3))
  }
  return(ee)
}
```

Elementary effects.

In [8]:

```
z.ee <- ee.oat(z.design, z.outputs.clean)
z.ee %>% dim
```

500 84 49

mu, mu*, and sigma.

In [9]:
```
z.mu <- apply(z.ee, 3, function(M){
  apply(M, 2, mean)
})
z.mu <- melt(
    cbind(
        data.table(Input=rownames(z.mu)),
        data.table(z.mu)
    ),
    id.vars="Input",
    variable.name = "Output",
    value.name = "mu"
)
z.mu %>% head
```

A data.table: 6 x 3

Input	Output	mu
<chr>	<fct>	<dbl>
advertising budget	bioproduct market share mass	0.303273857
advertising start time	bioproduct market share mass	-0.387986934
aversion to NPV deviation	bioproduct market share mass	-0.001418284
base external investor ask rate	bioproduct market share mass	-0.023685767
bioproduct long term price	bioproduct market share mass	-0.160606092
bioproduct offtake agreement	bioproduct market share mass	0.512129831

In [10]:

```r
z.mu.star <- apply(abs(z.ee), 3, function(M){
  apply(M, 2, mean)
})
z.mu.star <- melt(
    cbind(
        data.table(Input=rownames(z.mu.star)),
        data.table(z.mu.star)
    ),
    id.vars="Input",
    variable.name = "Output",
    value.name = "mu*"
)
z.mu.star %>% head
```

A data.table: 6 x 3

Input	Output	mu*
<chr>	<fct>	<dbl>
advertising budget	bioproduct market share mass	0.31027692
advertising start time	bioproduct market share mass	0.41195148
aversion to NPV deviation	bioproduct market share mass	0.01152693
base external investor ask rate	bioproduct market share mass	0.26447606
bioproduct long term price	bioproduct market share mass	0.39980623
bioproduct offtake agreement	bioproduct market share mass	0.77572582

In [11]:
```
z.sigma <- apply(z.ee, 3, function(M){
  apply(M, 2, sd)
})
z.sigma <- melt(
    cbind(
        data.table(Input=rownames(z.sigma)),
        data.table(z.sigma)
    ),
    id.vars="Input",
    variable.name = "Output",
    value.name = "sigma"
)
z.sigma %>% head
```

A data.table: 6 x 3

Input	Output	sigma
<chr>	<fct>	<dbl>
advertising budget	bioproduct market share mass	1.1085135
advertising start time	bioproduct market share mass	1.4664525
aversion to NPV deviation	bioproduct market share mass	0.1115146
base external investor ask rate	bioproduct market share mass	0.6278341
bioproduct long term price	bioproduct market share mass	1.7387415
bioproduct offtake agreement	bioproduct market share mass	4.5942430

In [12]:
```
z.results <- merge(merge(z.mu, z.mu.star, on=c("Input", "Output")), z.si
gma, on=c("Input", "Output"))
z.results %>% head
```

A data.table: 6 x 5

Input	Output	mu	mu*	sigma
<chr>	<fct>	<dbl>	<dbl>	<dbl>
advertising budget	bioproduct market share mass	3.032739e-01	3.102769e-01	1.108513e+00
advertising budget	current market size economic	-2.412707e+11	9.415770e+11	4.866741e+12
advertising budget	current market size mass	0.000000e+00	0.000000e+00	0.000000e+00
advertising budget	long term market share	-6.742848e-04	1.255821e-03	1.507683e-02
advertising budget	long term market value	-3.252294e+08	3.252294e+08	7.252179e+09
advertising budget	Adopters	4.651019e+07	5.092098e+07	1.587423e+08

```
In [13]: z.results[, `:=`(
    `mu rank`    = frank(- `mu`    ),
    `mu* rank`   = frank(- `mu*`   ),
    `sigma rank` = frank(- `sigma`)
), by=.(Output)]
z.results %>% head
```

A data.table: 6 x 8

Input	Output	mu	mu*	sigma	mu rank	mu* rank	sigma rank
<chr>	<fct>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
advertising budget	bioproduct market share mass	3.032739e-01	3.102769e-01	1.108513e+00	3	15	24
advertising budget	current market size economic	-2.412707e+11	9.415770e+11	4.866741e+12	80	23	25
advertising budget	current market size mass	0.000000e+00	0.000000e+00	0.000000e+00	43	44	44
advertising budget	long term market share	-6.742848e-04	1.255821e-03	1.507683e-02	64	49	51
advertising budget	long term market value	-3.252294e+08	3.252294e+08	7.252179e+09	71	30	24
advertising budget	Adopters	4.651019e+07	5.092098e+07	1.587423e+08	6	13	23

Interpret results.

Interpretations:

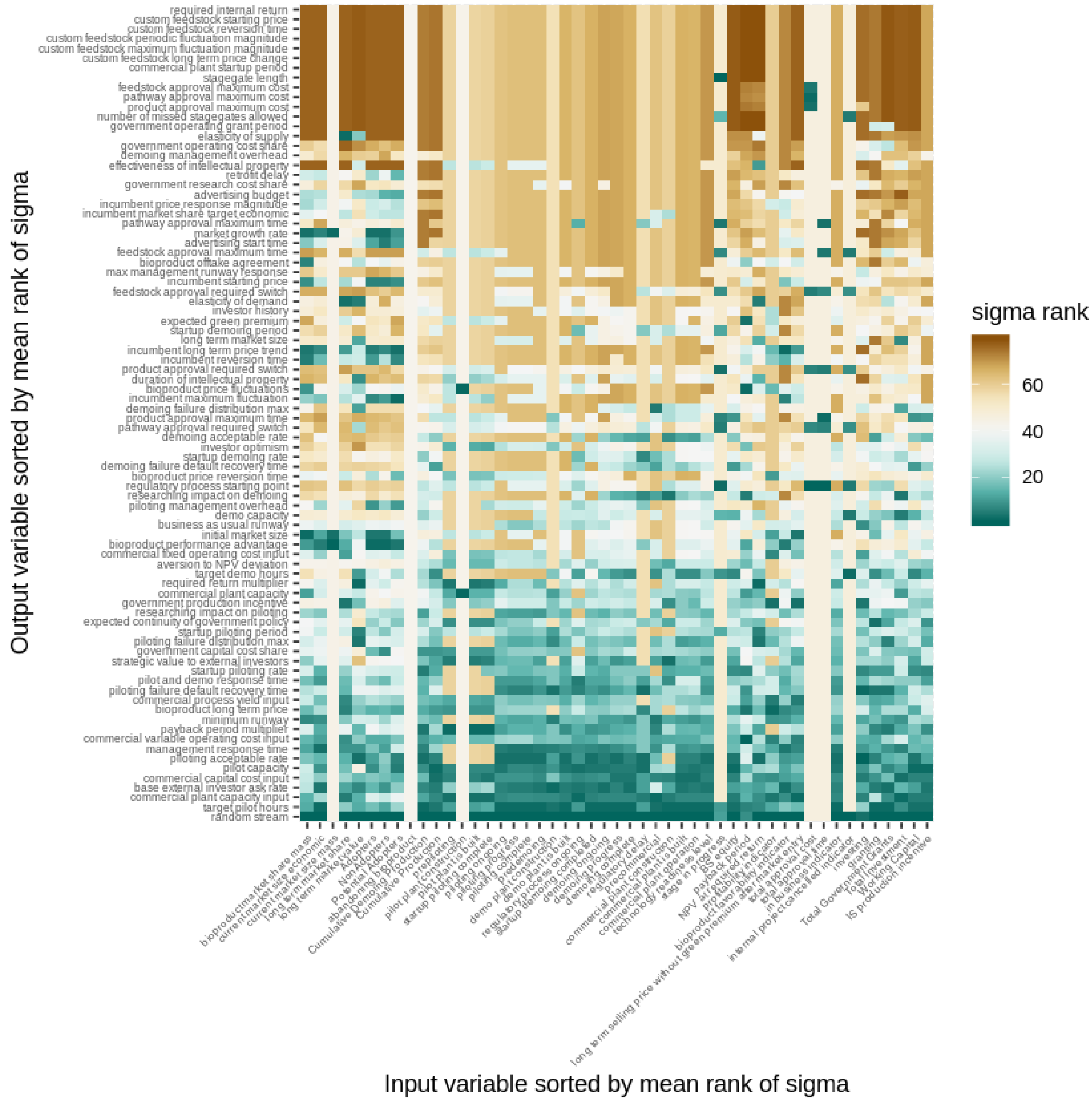
- mu: influence of variable
- mustar: influence of variable, accounting for non-monoticity
- sigma: non-linear and interaction effects for variable

Heat map for mu*.

In [14]:
```
ggplot(
    z.results,
    aes(
        x=factor(
            Output,
            levels=z.results[, .(`sort` = mean(`mu* rank`)), by=.(Output
)][order(sort), `Output`]
        ),
        y=factor(
            Input,
            levels=z.results[, .(`sort` = mean(`mu* rank`)), by=.(Input)
][order(sort), `Input`]
        ),
        fill=`mu* rank`
    )
) +
    geom_tile() +
    scale_fill_distiller(type="div") +
    xlab("Input variable sorted by mean rank of mu*") +
    ylab("Output variable sorted by mean rank of mu*") +
    theme(
        axis.text=element_text(size=5),
        axis.text.x = element_text(angle = 45, hjust=1)
    )
```

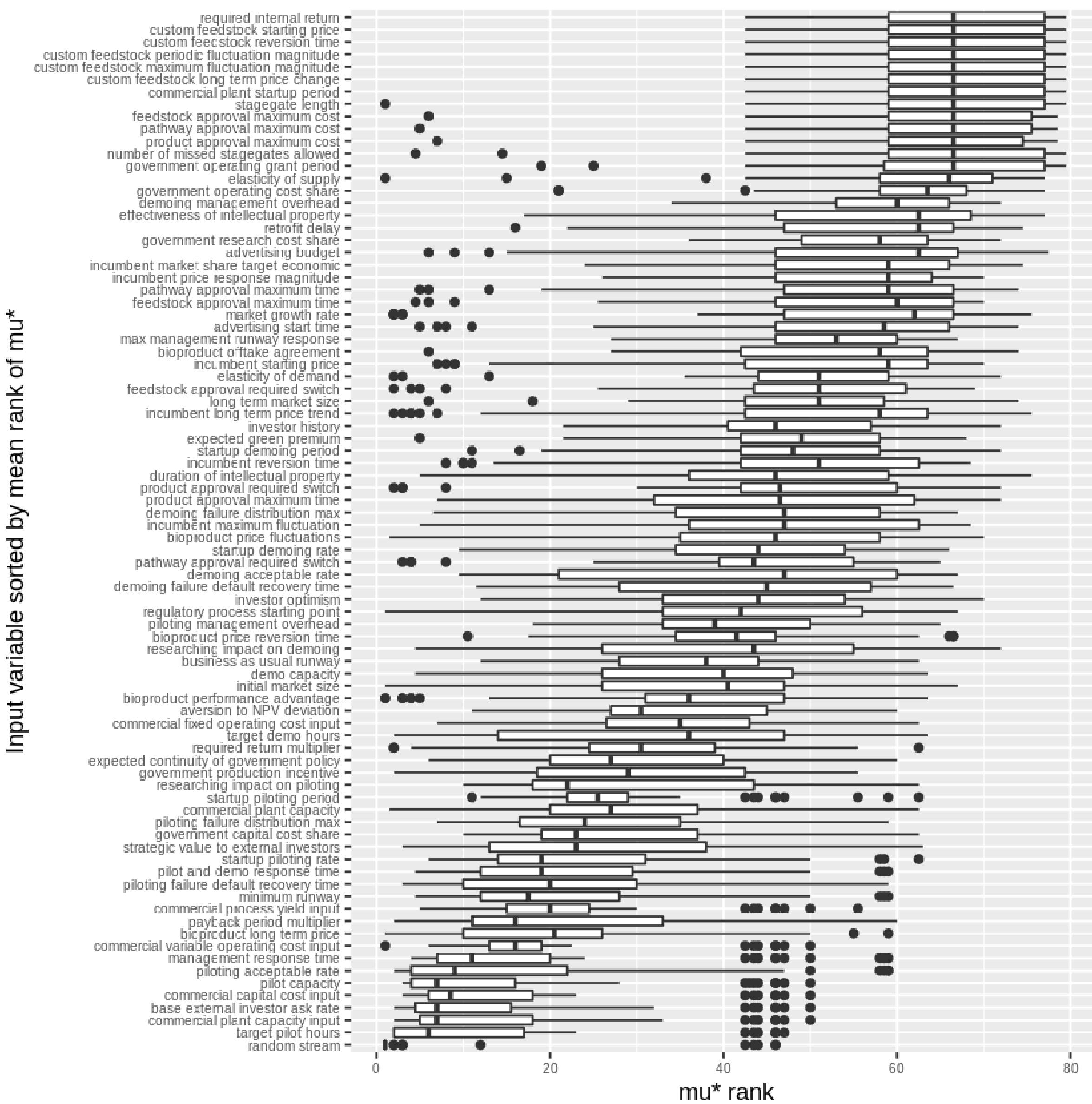
Heat map for sigma.

In [15]:
```
ggplot(
    z.results,
    aes(
        x=factor(
            Output,
            levels=z.results[, .(`sort` = mean(`sigma rank`)), by=.(Outp
ut)][order(sort), `Output`]
        ),
        y=factor(
            Input,
            levels=z.results[, .(`sort` = mean(`sigma rank`)), by=.(Inpu
t)][order(sort), `Input`]
        ),
        fill=`sigma rank`
    )
) +
    geom_tile() +
    scale_fill_distiller(type="div") +
    xlab("Input variable sorted by mean rank of sigma") +
    ylab("Output variable sorted by mean rank of sigma") +
    theme(
        axis.text=element_text(size=5),
        axis.text.x = element_text(angle = 45, hjust=1)
    )
```



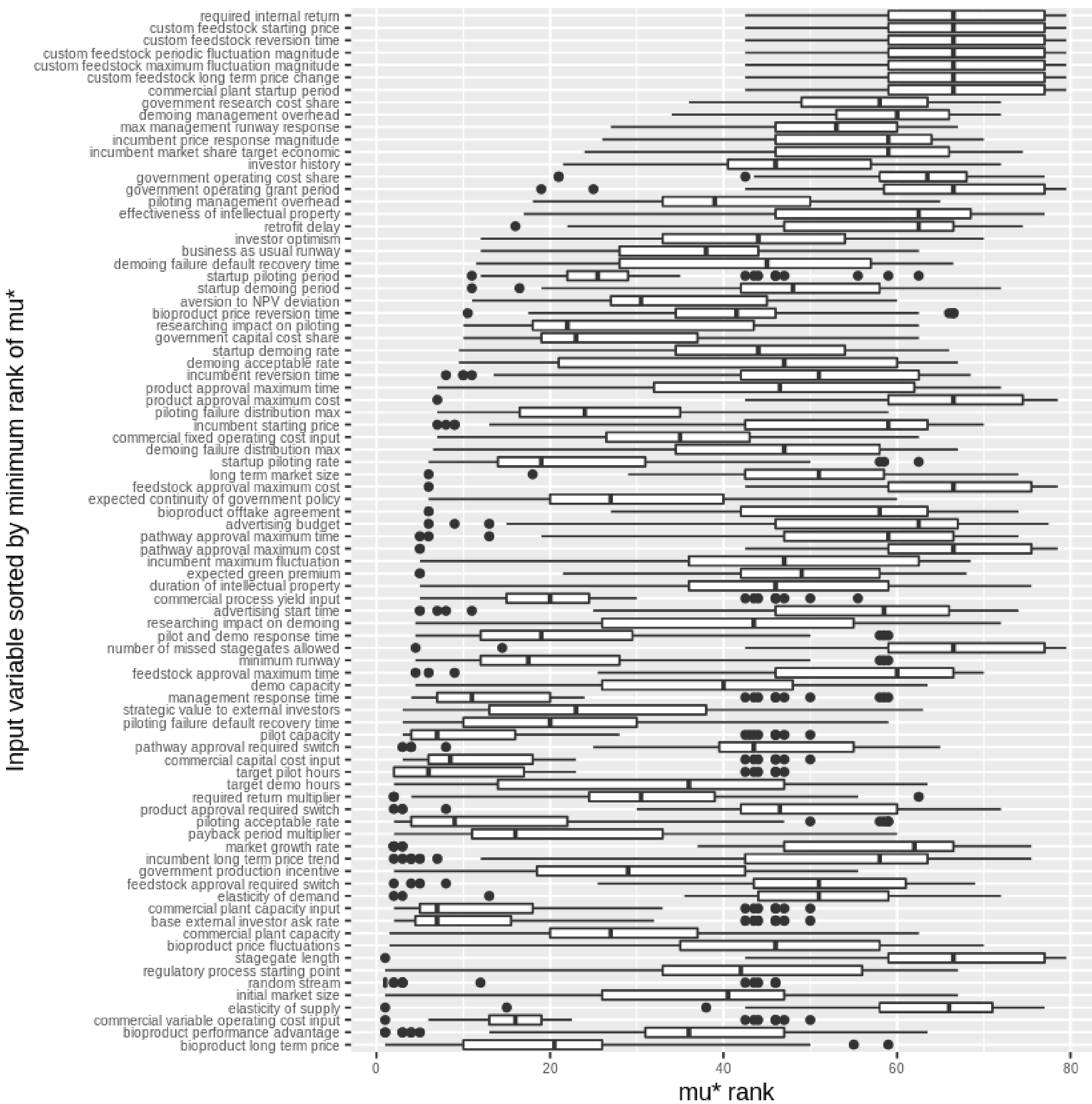
Box plots of ranks for mu*.

```
In [16]: ggplot(
    z.results,
    aes(
        x=factor(
            Input,
            levels=z.results[, .(`sort` = mean(`mu* rank`)), by=.(Input)
][order(sort), `Input`]
        ),
        y=`mu* rank`
    )
) +
    geom_boxplot() +
    coord_flip() +
    xlab("Input variable sorted by mean rank of mu*") +
    theme(axis.text=element_text(size=6))
```



In [17]:

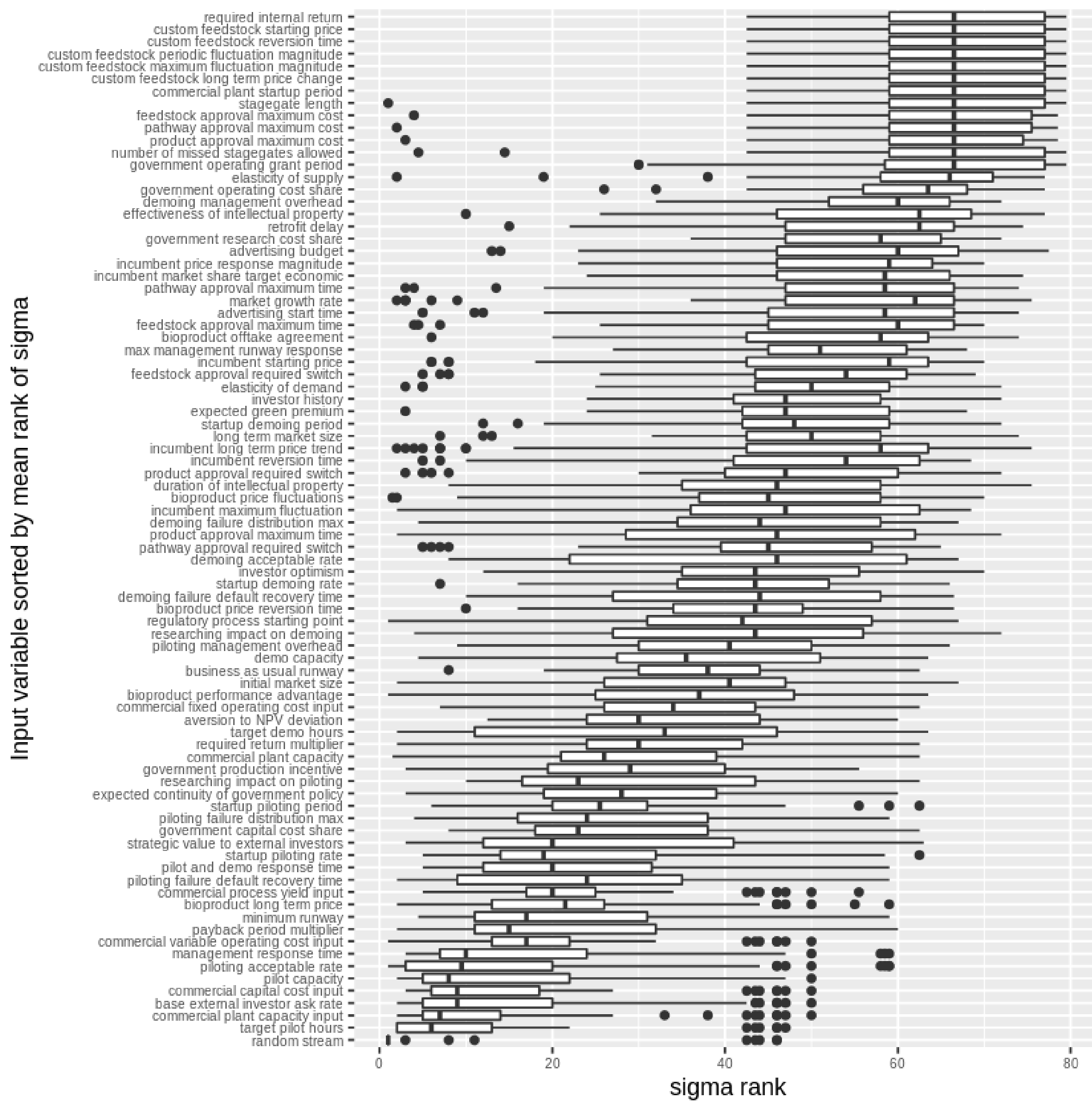
```
ggplot(
    z.results,
    aes(
        x=factor(
            Input,
            levels=z.results[, .(`sort` = min(`mu* rank`)), by=.(Input)]
[order(sort), `Input`]
        ),
        y=`mu* rank`
    )
) +
    geom_boxplot() +
    coord_flip() +
    xlab("Input variable sorted by minimum rank of mu*") +
    theme(axis.text=element_text(size=6))
```



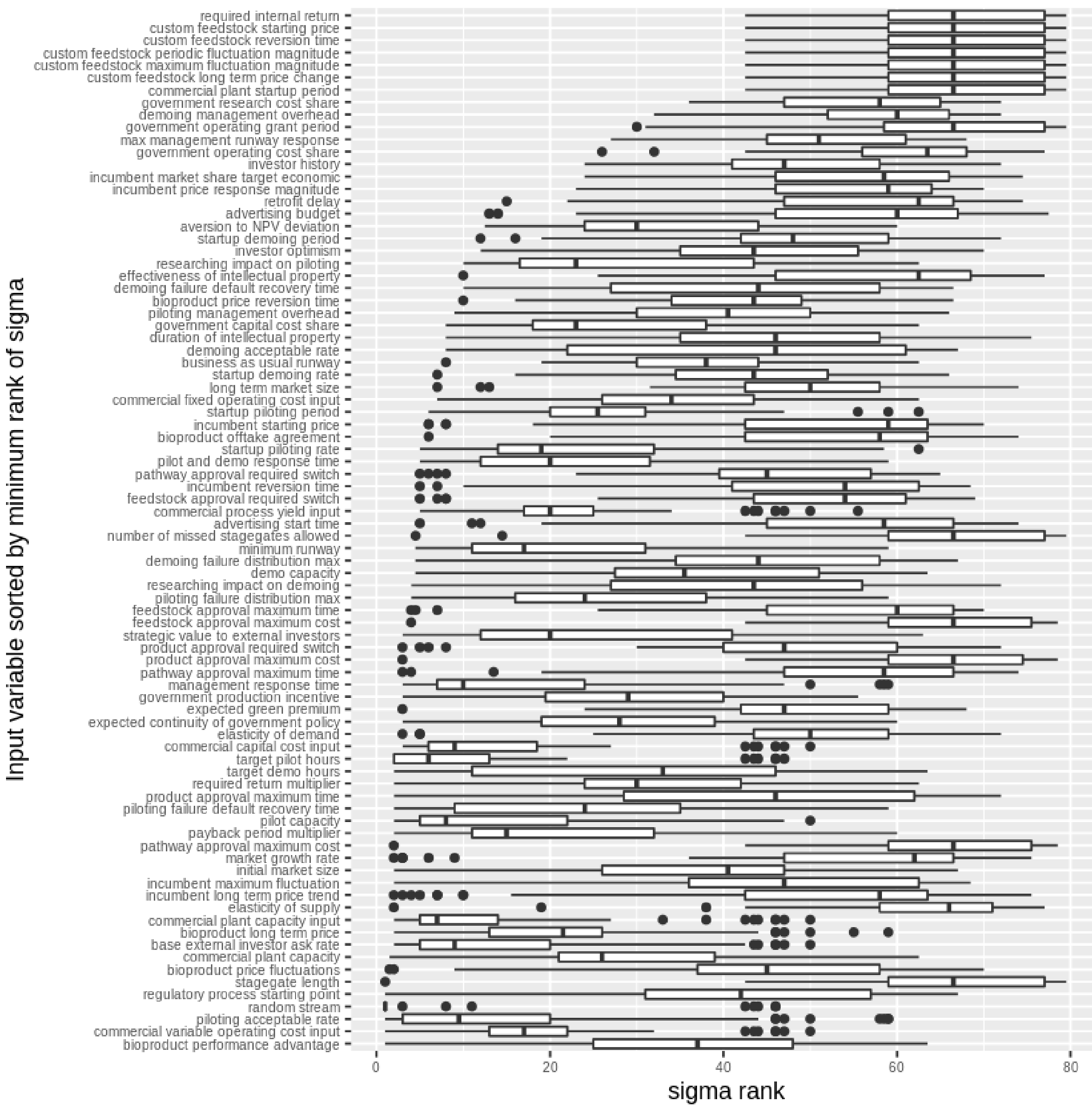
Box plots of ranks for sigma.

In [18]:

```
ggplot(
    z.results,
    aes(
        x=factor(
            Input,
            levels=z.results[, .(`sort` = mean(`sigma rank`)), by=.(Inpu
t)][order(sort), `Input`]
        ),
        y=`sigma rank`
    )
) +
    geom_boxplot() +
    coord_flip() +
    xlab("Input variable sorted by mean rank of sigma") +
    theme(axis.text=element_text(size=6))
```



```
In [19]: ggplot(
    z.results,
    aes(
        x=factor(
            Input,
            levels=z.results[, .(`sort` = min(`sigma rank`)), by=.(Input
)][order(sort), `Input`]
        ),
        y=`sigma rank`
    )
) +
    geom_boxplot() +
    coord_flip() +
    xlab("Input variable sorted by minimum rank of sigma") +
    theme(axis.text=element_text(size=6))
```



Select inputs variables for variance-based sensitivity analysis.

Select the variables whose median rank is less than 30.

In [20]:

```
z.results[,
    .(
        `mu* minimum rank`    = min   (`mu* rank`),
        `mu* mean rank`       = mean  (`mu* rank`),
        `mu* median rank`     = median(`mu* rank`),
        `sigma minimum rank` = min   (`sigma rank`),
        `sigma mean rank`     = mean  (`sigma rank`),
        `sigma median rank`   = median(`sigma rank`)
    ),
    by=.(Input)
][order(`mu* minimum rank`)][
    `mu* median rank` <= 30 | `sigma median rank` <= 30
]
```

A data.table: 26 x 7

Input	mu* minimum rank	mu* mean rank	mu* median rank	sigma minimum rank	sigma mean rank	sigma median rank
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
bioproduct long term price	1.0	21.989796	20.5	2.0	22.938776	21.5
commercial variable operating cost input	1.0	19.183673	16.0	1.0	20.000000	17.0
random stream	1.0	5.816327	1.0	1.0	5.816327	1.0
commercial plant capacity	1.5	28.193878	27.0	1.5	29.418367	26.0
base external investor ask rate	2.0	13.744898	7.0	2.0	14.755102	9.0
commercial plant capacity input	2.0	13.673469	7.0	2.0	14.428571	7.0
government production incentive	2.0	28.897959	29.0	3.0	29.295918	29.0
payback period multiplier	2.0	22.020408	16.0	2.0	21.285714	15.0
piloting acceptable rate	2.0	17.908163	9.0	1.0	17.826531	9.5
required return multiplier	2.0	30.326531	30.5	2.0	30.285714	30.0
target pilot hours	2.0	11.948980	6.0	2.0	11.775510	6.0
commercial capital cost input	3.0	14.326531	8.5	3.0	15.173469	9.0
pilot capacity	3.0	14.632653	7.0	2.0	15.285714	8.0
piloting failure default recovery time	3.0	23.000000	20.0	2.0	23.857143	24.0
strategic value to external investors	3.0	26.132653	23.0	3.0	26.265306	20.0
management response time	4.0	18.091837	11.0	3.0	18.897959	10.0
minimum runway	4.5	22.683673	17.5	4.5	22.785714	17.0
pilot and demo response time	4.5	23.581633	19.0	5.0	24.030612	20.0
commercial process yield input	5.0	22.500000	20.0	5.0	22.938776	20.0
expected continuity of government policy	6.0	29.071429	27.0	3.0	28.653061	28.0
startup piloting rate	6.0	24.122449	19.0	5.0	24.377551	19.0
piloting failure distribution max	7.0	27.571429	24.0	4.0	27.397959	24.0
government capital cost share	10.0	27.520408	23.0	8.0	27.357143	23.0
researching impact on piloting	10.0	28.571429	22.0	10.0	28.755102	23.0

Input	mu* minimum rank	mu* mean rank	mu* median rank	sigma minimum rank	sigma mean rank	sigma median rank
<chr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
aversion to NPV deviation	11.0	34.142857	30.5	12.5	33.285714	30.0
startup piloting period	11.0	28.459184	25.5	6.0	28.020408	25.5