Hoe ik de relaties heb vereenvoudigd:

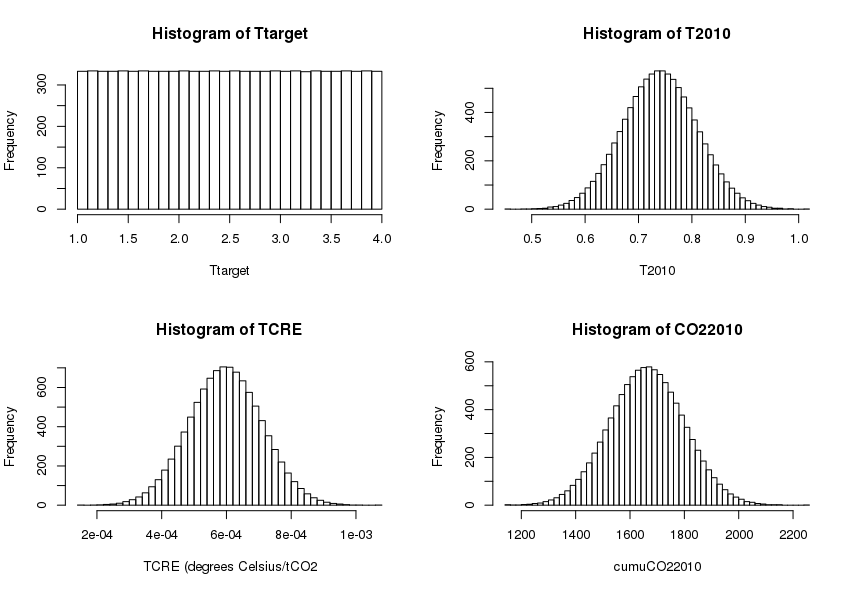
De volgende parameters zijn gesampled met LHS:

* Ttarget1-4
* T2010
* TCRE
* CO22010
* Costs.slope
* baselineCO2 (constant op 6000 GtCO2)

Parameters met normale verdeling:

* T2010
* TCRE
* CO22010
* Costs.slope

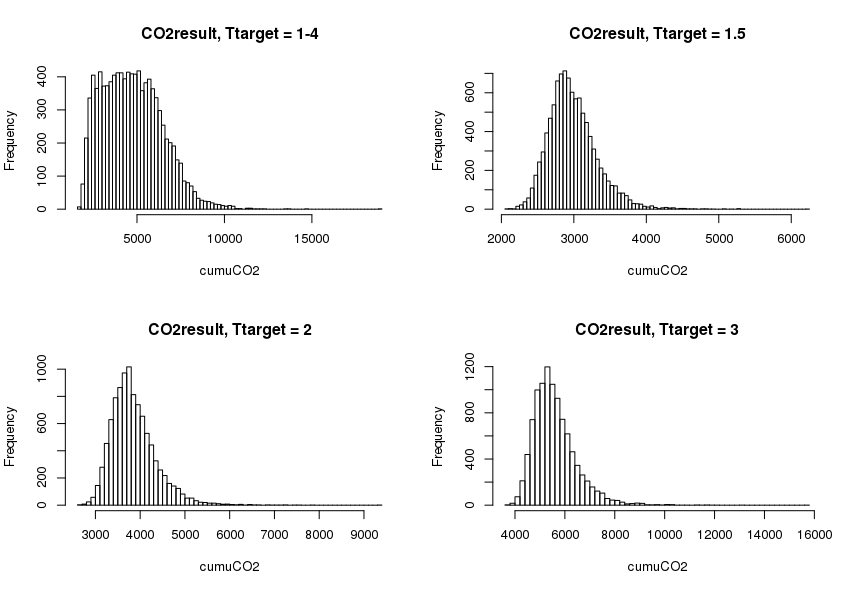
Histogrammen van inputparameters:

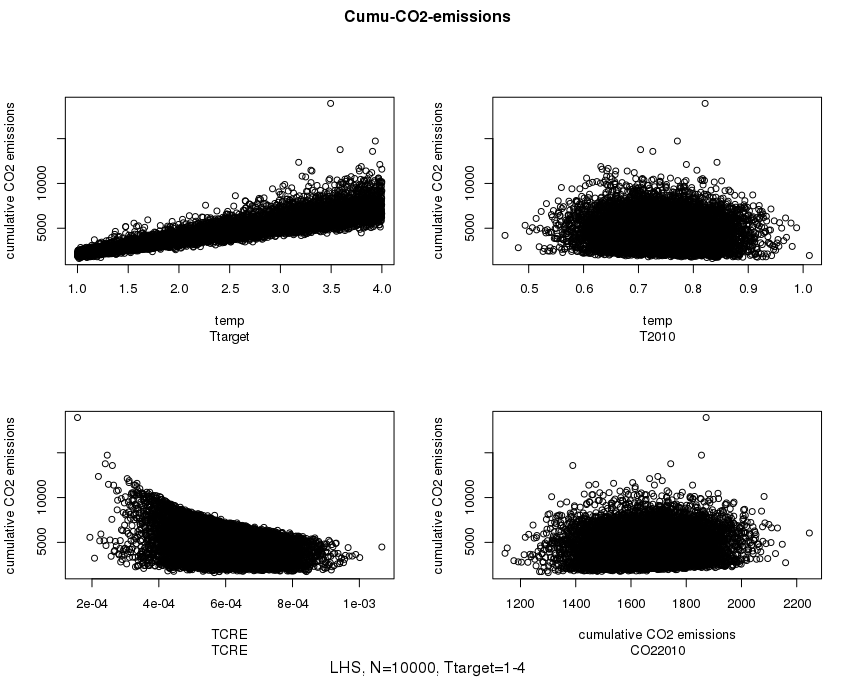


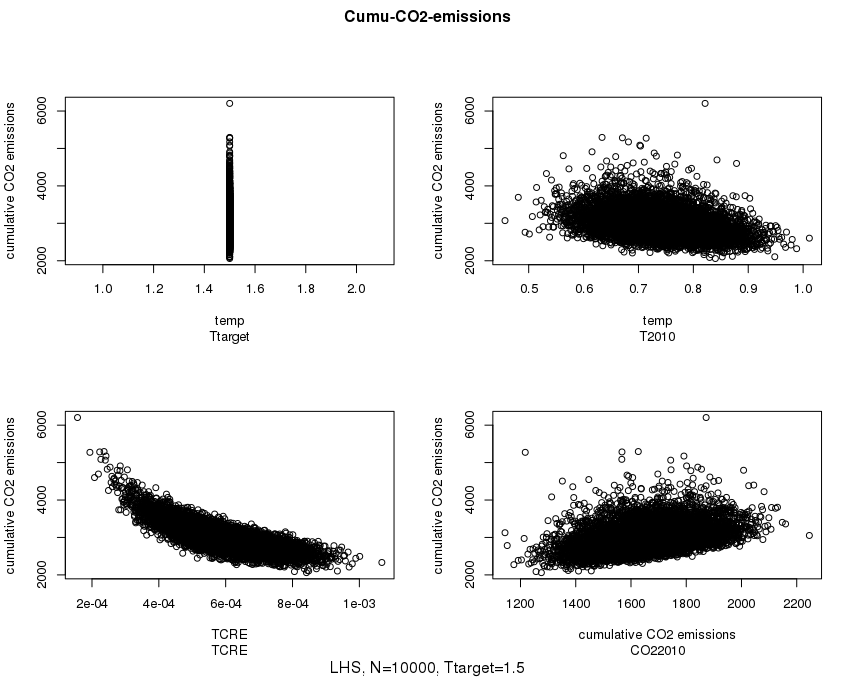
Eerst is een carbon-budget uitgerekend met de volgende formule voor elke input:

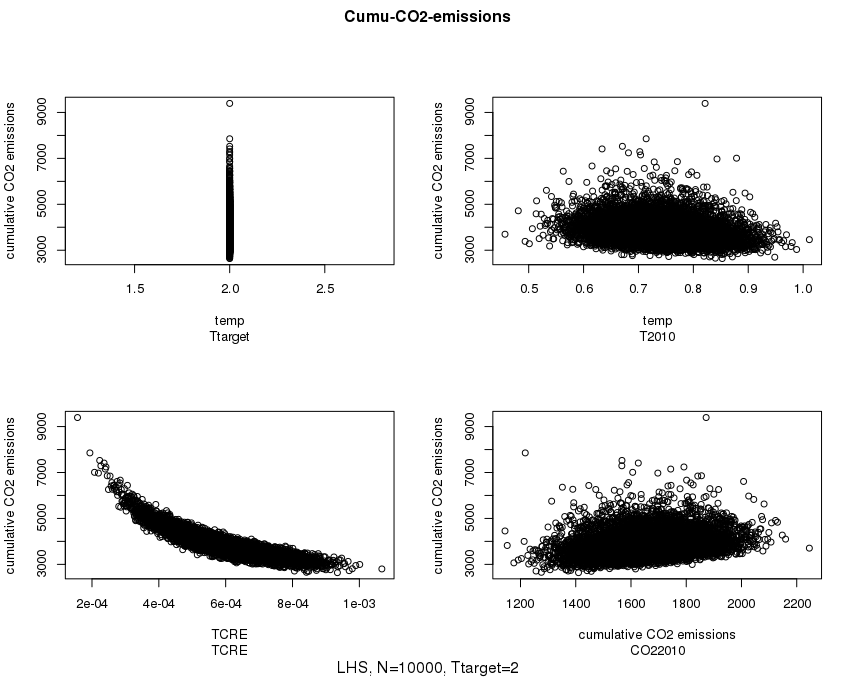
cumu-CO2result = cumu-CO22010 + (Ttarget - T2010)/TCRE

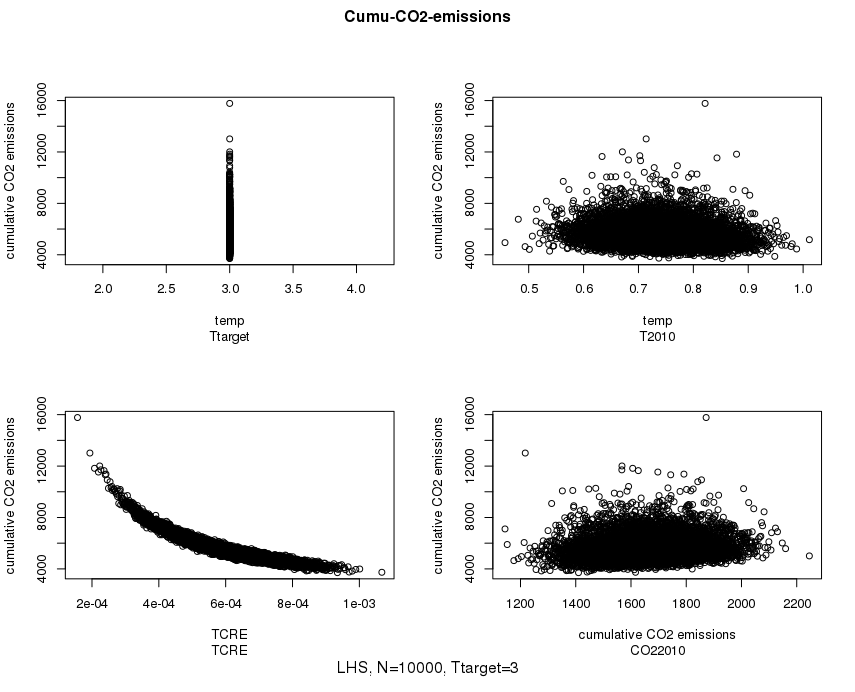
Ttarget in gevarieerd in de volgende waarden: 1-4 (continu verdeeld), 1.5, 2 en 3.

Histogram van cumu-CO2result:

De volgende scatterplots geven aan hoe de input invloed heeft op de output cumuCO2result:







Conclusies: TCRE heeft veel invloed.

Vervolgens zijn de kosten uitgerekend met behulp van de volgende formule:

Costs = costs.slope \* (cumuCO2 – baselineCO2)

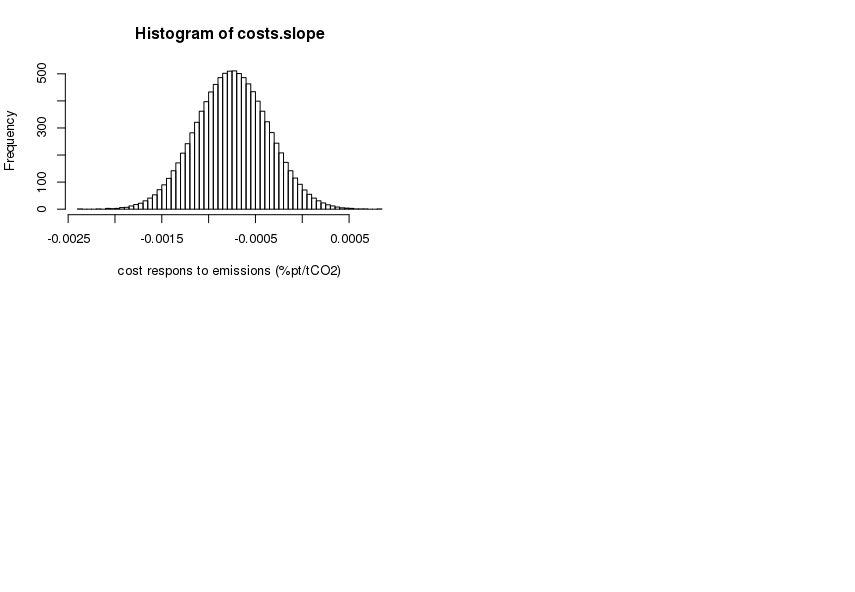
Aannames in dit verband zijn:

* Er is een lineair verband tussen kosten en cumulatieve CO2 uitstoot:

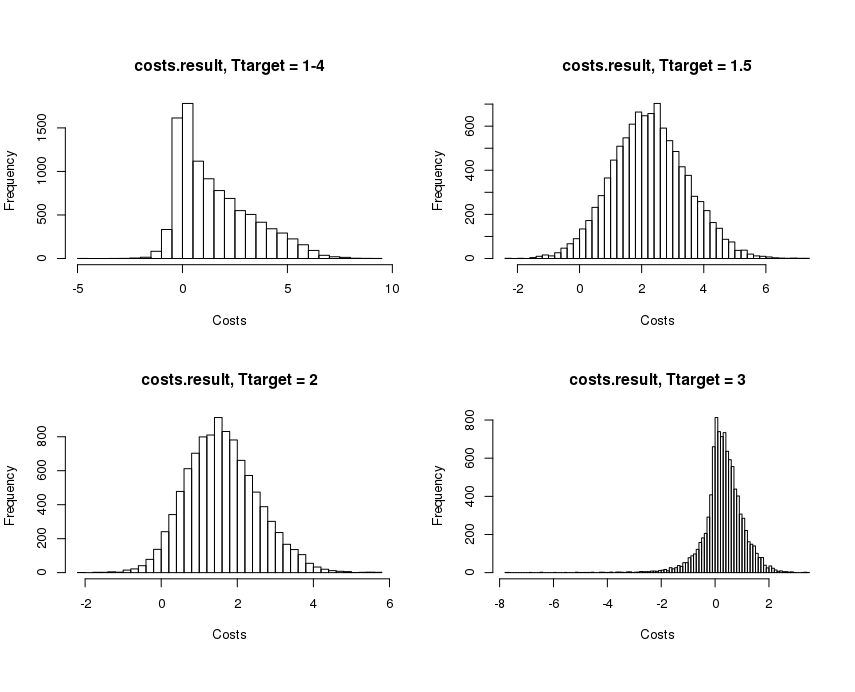
Costs = costs.slope \* cumuCO2 + costs.intercept

* In het baseline scenario investeren we niet in mitigatiemaatregelen en hebben we dus geen investeringskosten, dus als Costs = 0, dan cumuCO2 = baselineCO2, daaruit volgt:

costs.intercept = -1 \* baselineCO2 \* cost.slope

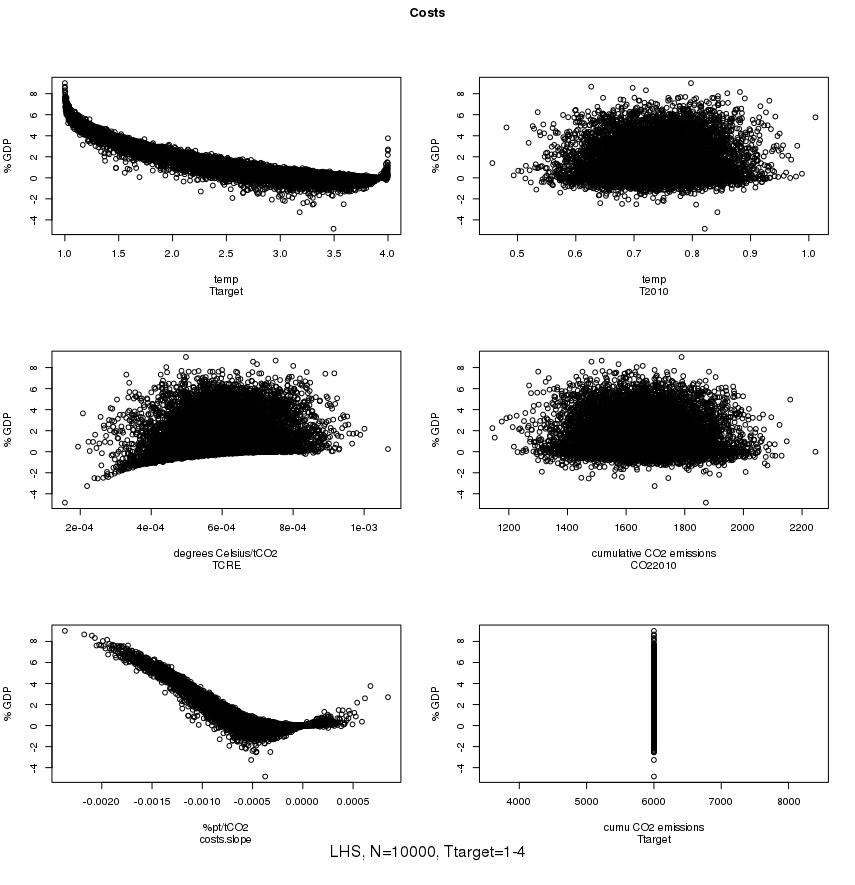


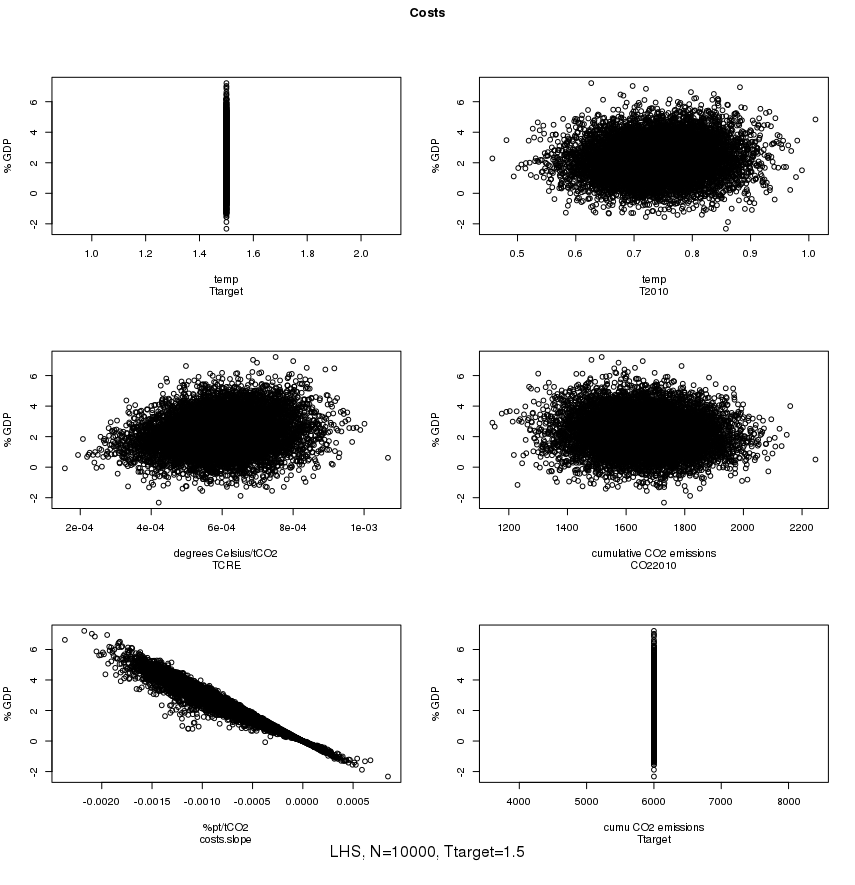
Histogram van costs.slope:

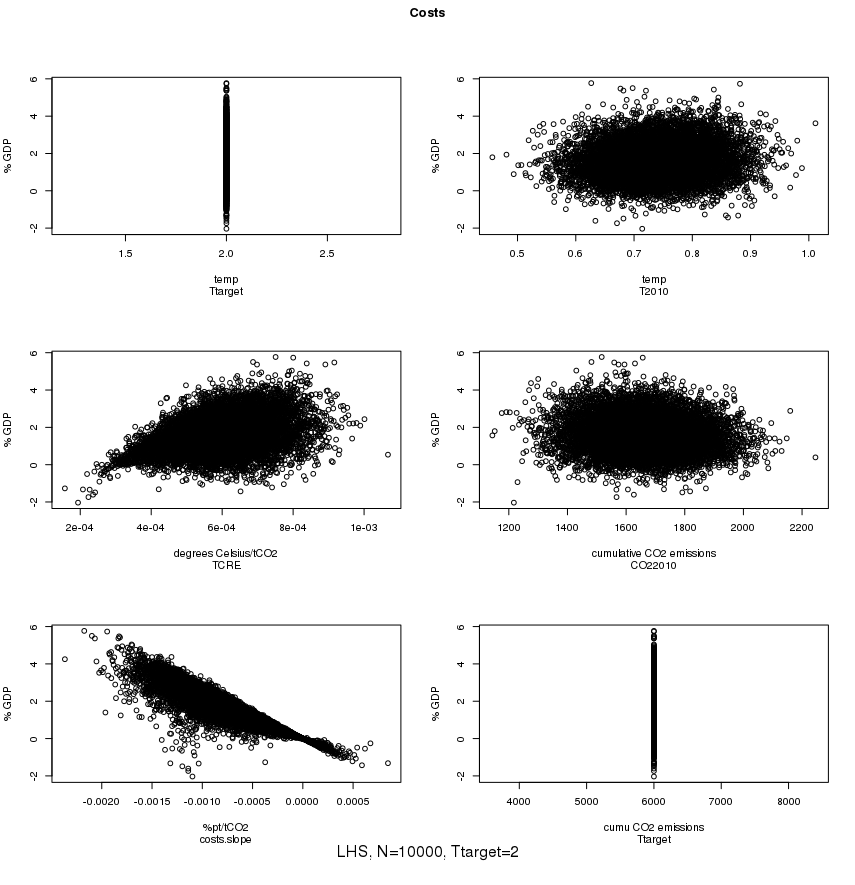


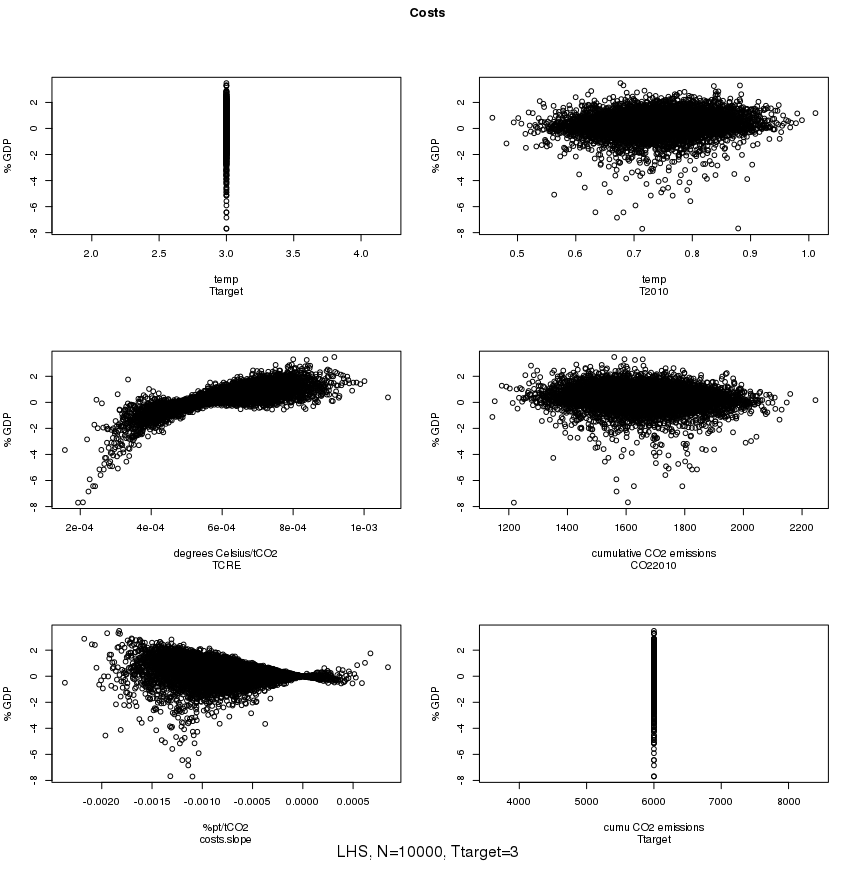
Histogram van costs:

Scatterplots:









We zien dat bij Ttarget = 3 TCRE meer invloed heft dan costs.slope, terwijl bij andere Ttarget costs.slope meer invleod heeft.

cumuCO2result uitgezet tegen costs:

