Initialization

In [ ]:
```
using LinearAlgebra
using Plots

include("lsq_classifier_data.jl");
```

In [ ]:
```
beta = X'\y;
```

In [ ]:
```
y_hat = X' * beta
y_test_hat = X_test' * beta;
```

In [ ]:
```
function compute_loss(y_hat,y)
    match = 0
    for i in 1:size(y_hat)[1]
        if sign.(y_hat[i]) != y[i]
            match+=1
        end
    end
    return match/size(y_hat)[1]
end
```

Out[ ]:
```
compute_loss (generic function with 1 method)
```

In [ ]:
```
loss_train = compute_loss(y_hat,y)
```

Out[ ]:
```
0.31
```

In [ ]:
```
loss_test = compute_loss(y_test_hat,y_test)
```

Out[ ]:
```
0.37
```

In [ ]:
```
lambda = 10 .^ range(-1, 4, length=100);
```

In [ ]:
```
test_out = []
train_out = []
```

Out[ ]:
```
Any[]
```

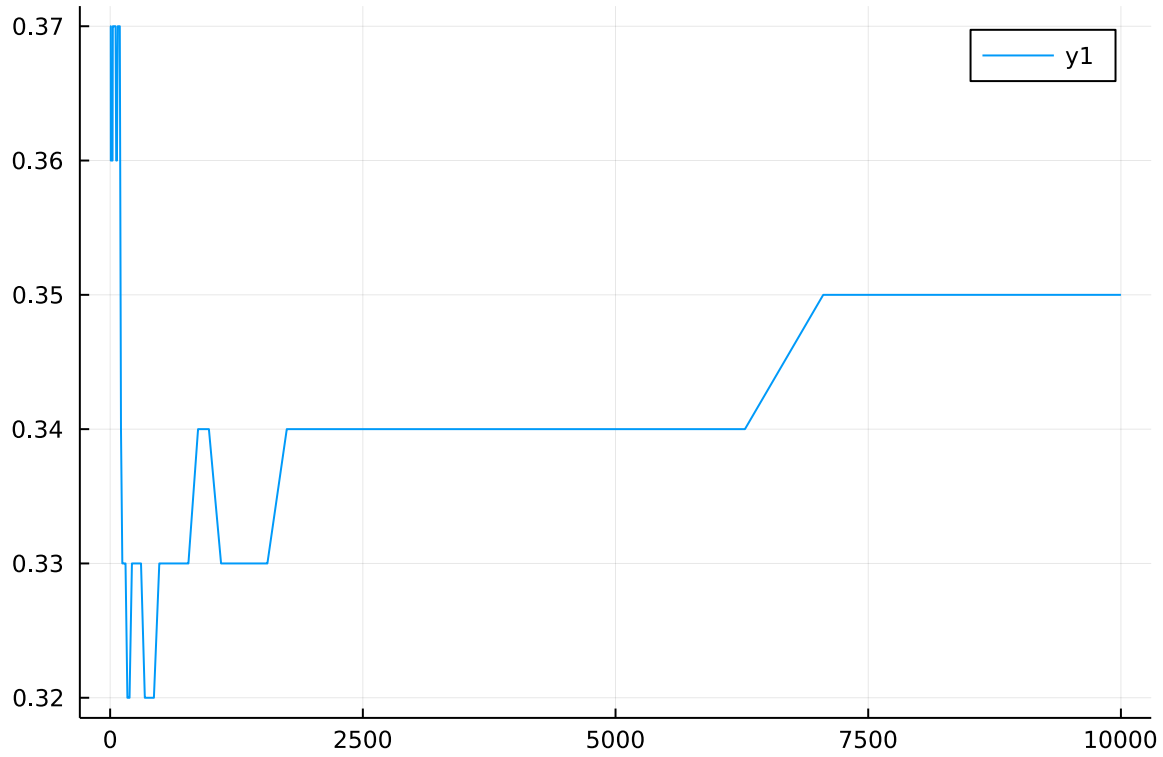
In [ ]:
```
for i in lambda
    beta = inv(X*(X') + i*Matrix{Float64}(I,50,50)) * X * y
    append!(test_out,compute_loss((X_test')*beta, y_test))
    append!(train_out,compute_loss((X')*beta, y))
end
```

In [ ]:
```
lambda[findmin(test_out)[2]]
```

Out[ ]:
```
170.73526474706904
```

In [ ]:
```
plot(lambda,test_out)
```

Out[ ]:



In [ ]:
```
plot(lambda,test_out)
```

Out[ ]: