Gaussian Process Simulations

Matt Dickenson mcd31
STA571/CS590.01 Due: 21 March, 2014

1 Code

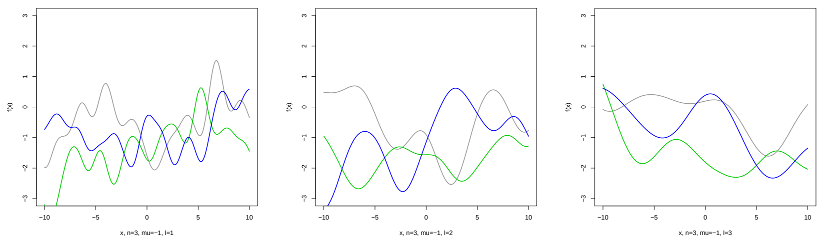
The following code was used to generate (function) samples from Guassian processes using several covariance functions. The simulations are shown below.

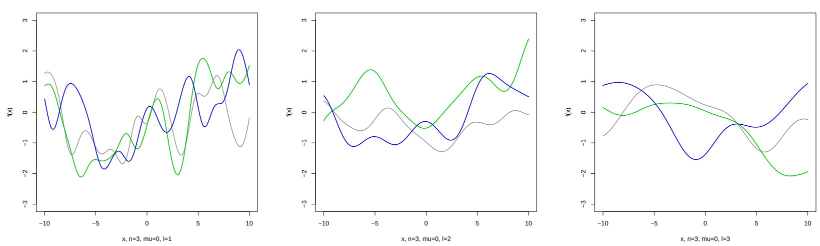
Listing 1: GP code

```
# CS 590.01
# Homework 17
# 21 March, 2014
# Matt Dickenson
# mcd31@duke.edu

# For the squared exponential and
# two other covariance functions (your choice)
# plot (function) samples from a Gaussian process.
# Several plots should be included for each covariance function,
# varying the number of draws, and the hyperparameters
# associated with each.

# set up workspace
rm(list=ls())
library(MASS)
set.seed(8675309)

# covariance_functions

# squared exponential covariance function
# try l=1,2,...
squared_expo = function(x1, x2, l=1){
  sigma = matrix(0, nrow=length(x1), ncol=length(x1))
  for (i in 1:nrow(sigma)) {
    for (j in 1:ncol(sigma)) {
      sigma[i,j] = exp( -( ( ( x1[i]-x2[j] )^2 )/ (2 *l^2) ) )
    }
  }
  return(sigma)
}

# rational quadratic covariance function
rational_quad = function(x1, x2, l=1, alpha=0.5) {
  sigma = matrix(0, nrow=length(x1), ncol=length(x1))
  for (i in 1:nrow(sigma)) {
    for (j in 1:ncol(sigma)) {
```

```
      r = abs(x1[i]-x2[j])
      sigma[i,j] = ( 1 + (r^2)/(2 * alpha * l^2) )^(-alpha)
    }
  }
  return(sigma)
}

# gamma exponential covariance function
gamma_expo = function(x1, x2, l, gamma=1){
  sigma = matrix(0, nrow=length(x1), ncol=length(x1))
  for (i in 1:nrow(sigma)) {
    for (j in 1:ncol(sigma)) {
      r = abs(x1[i]-x2[j])
      sigma[i,j] = exp( -(r/l)^gamma )
    }
  }
  return(sigma)
}

plot_gp = function(nsamps=3, mu=0, l=1, alpha=0.5, gamma=1, cvfun="squared_
  xs = seq(-10,10,length.out=200)
  xlab = paste("x, n=", nsamps, ", mu=", mu, ", l=", l, sep="")

  if(cvfun=="squared exponential"){
    sigma = squared_expo(xs, xs, l)
  } else if(cvfun=="rational quadratic"){
    sigma = rational_quad(xs, xs, l, alpha)
    xlab = paste(xlab, ", alpha=", alpha, sep="")
  } else if(cvfun=="gamma exponential"){
    sigma = gamma_expo(xs, xs, l, gamma)
    xlab = paste(xlab, ", gamma=", gamma, sep="")
  }


  # draw samples
  samps = matrix(rep(0,length(xs)*nsamps), ncol=nsamps)
  for (i in 1:nsamps) {
    samps[,i] <- mvrnorm(1, rep(mu, length(xs)), sigma)
  }
  samps <- cbind(x=xs,as.data.frame(samps))

  # plot function draws
  plot(samps$x, samps$V1,
```

```
    lwd=2,
    ylim=c(-3, 3),
    type="l",
    col="grey60",
    xlab=xlab,
    ylab="f(x)")
  for(i in 3:(nsamps+1)) {
    lines(samps[,1], samps[,i], col=i, lwd=2)
  }
}


for(n in c(3, 5, 10)){
  for(m in c(-1, 0, 1)){
    for(l in c(1, 2, 3)){
      for(cv in c("squared_exponential", "rational_quadratic", "gamma_expon
        if(cv=="squared_exponential"){
          filename = paste("n", n, "-m", m, "-l", l, sep="")
          filename = paste(filename, ".pdf", sep="")
          pdf(filename)
            plot_gp(nsamps=n, mu=m, l=l, cvfun=cv)
          dev.off()
        } else if(cv=="rational_quadratic"){
          for(a in c(0.5, 2, 3)){
            filename = paste("n", n, "-m", m, "-l", l, sep="")
            filename = paste(filename, "-a", a*2, ".pdf", sep="")
            pdf(filename)
              plot_gp(nsamps=n, mu=m, l=l, alpha=a, cvfun=cv)
            dev.off()
          }
        } else if(cv=="gamma_exponential"){
          for(g in c(0.5, 1, 2)){
            filename = paste("n", n, "-m", m, "-l", l, sep="")
            filename = paste(filename, "-g", g*2, ".pdf", sep="")
            pdf(filename)
              plot_gp(nsamps=n, mu=m, l=l, gamma=g, cvfun=cv)
            dev.off()
          }
        }
      }
    }
  }
}
```



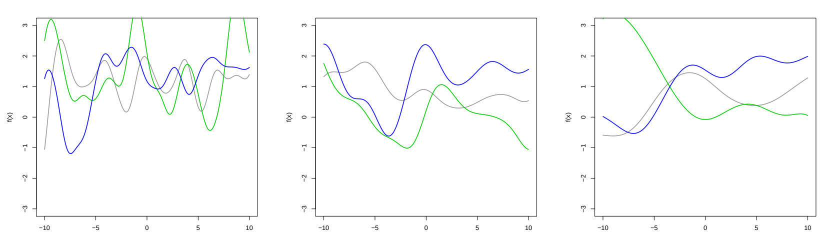


2 Squared Exponential Covariance

The simulations in this section use the squared exponential covariance function:

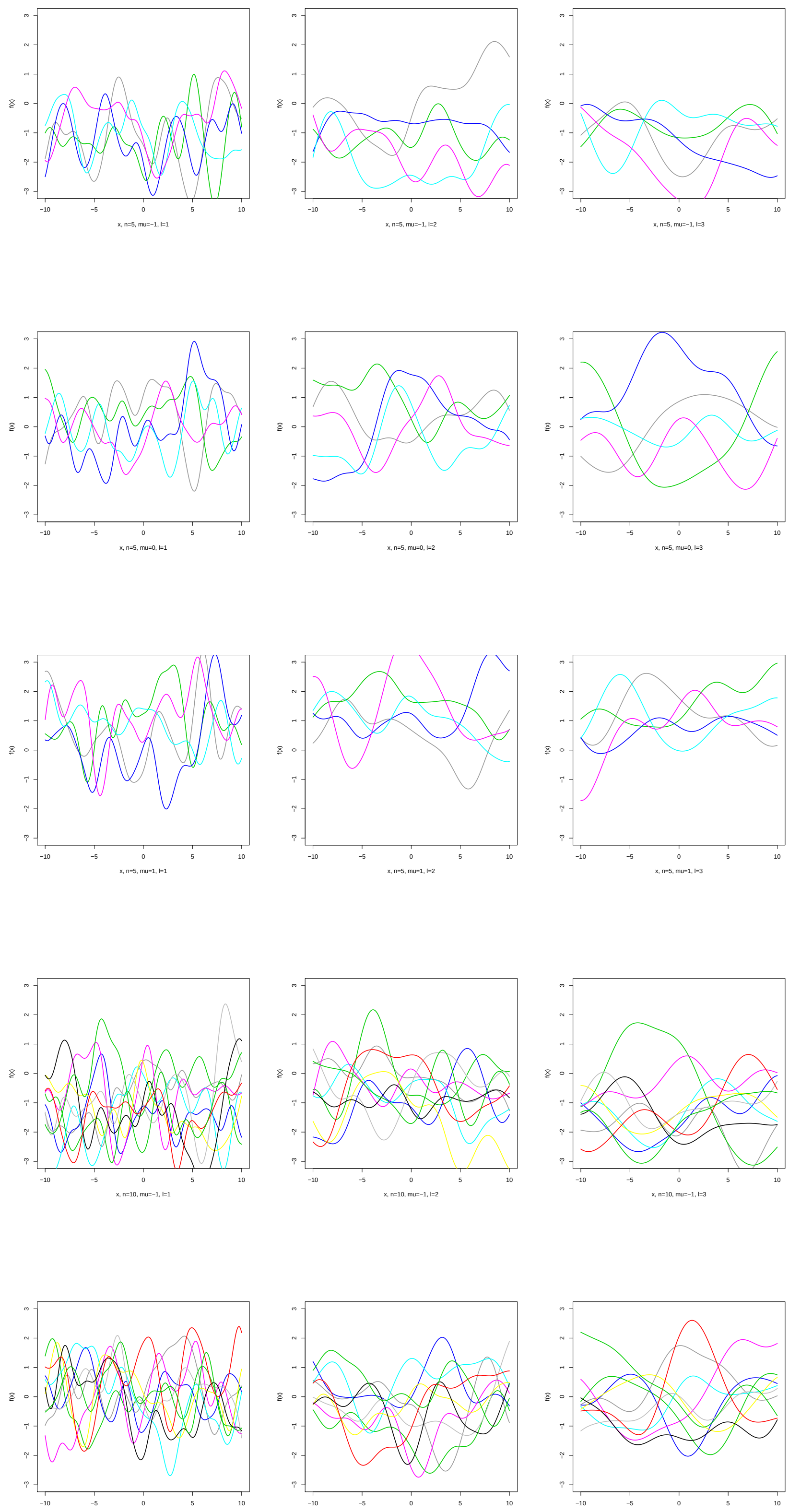
$$k(x, x') \quad = \quad \exp\{-\frac{(x-x')^2}{2\ell^2}\}$$

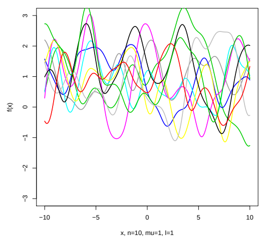
The parameters of interest here are the number of functions generated, n, the mean of the multivariate normal used for the simulations, μ, and the length scale ℓ. Each simulation plot shows the parameters used.

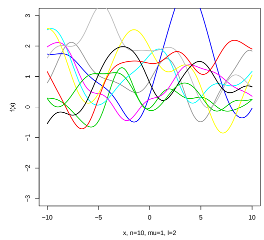


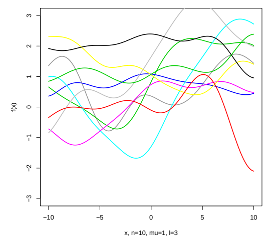
Gaussian Process Simulations

Matt Dickenson `mcd31`
STA571/CS590.01
Due: 21 March, 2014









3 Rational Quadratic Covariance

The simulations in this section use the rational quadratic covariance function:

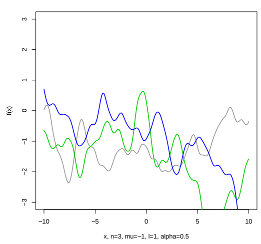
$$k(x, x') \quad = \quad (1 + \frac{(x - x')^2}{2\alpha\ell^2})^{-\alpha}$$

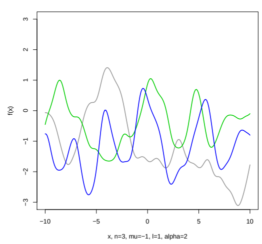
The parameters of interest here are the same as above with the addition of α. The tuple (α, ℓ) can be seen as a scale mixture.

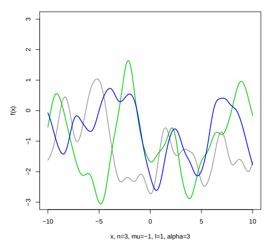
Gaussian Process Simulations

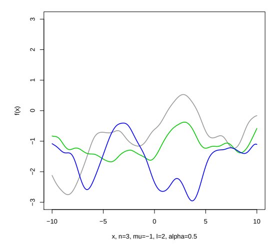
Matt Dickenson `mcd31`

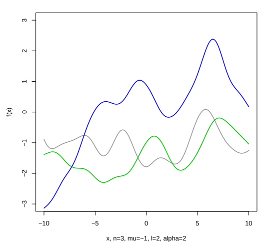
STA571/CS590.01 Due: 21 March, 2014

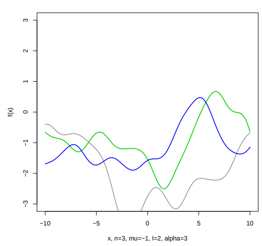


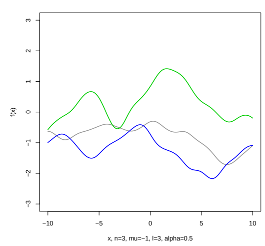


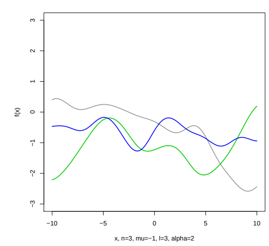


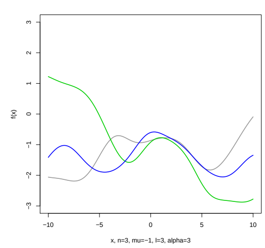








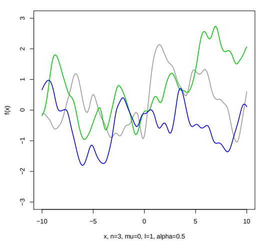


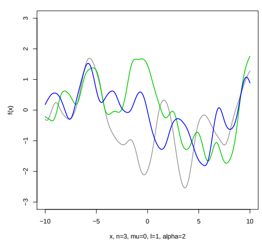


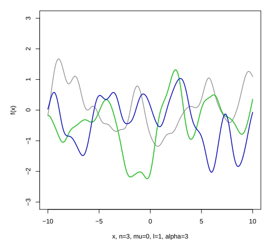
Gaussian Process Simulations

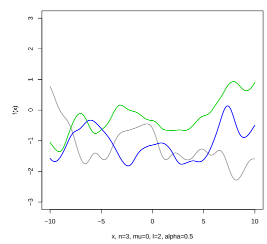
Matt Dickenson `mcd31`

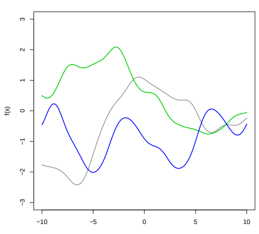
STA571/CS590.01 Due: 21 March, 2014

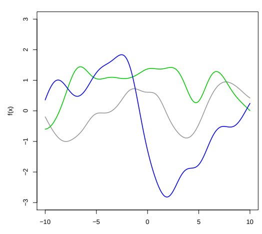


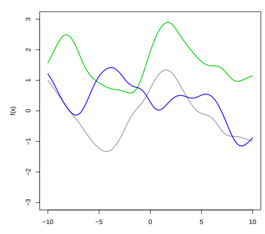


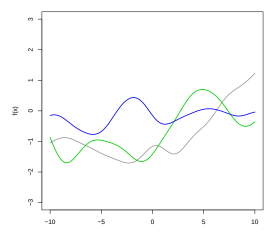


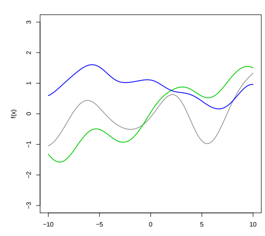






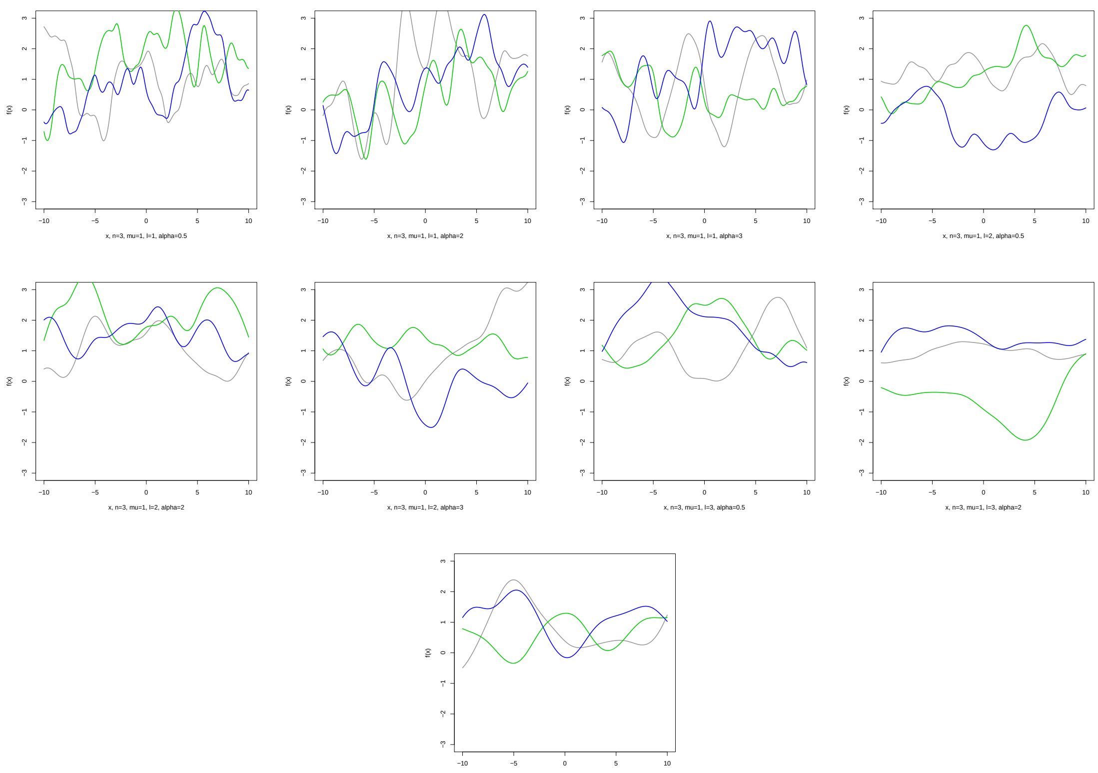






Gaussian Process Simulations

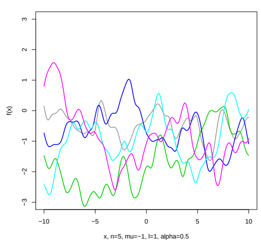
Matt Dickenson `mcd31`
STA571/CS590.01 Due: 21 March, 2014

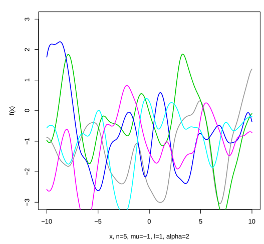


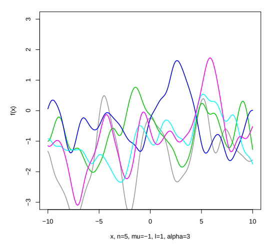
Gaussian Process Simulations

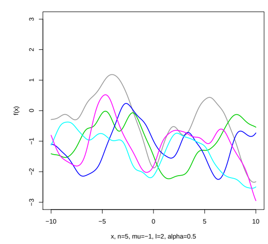
Matt Dickenson `mcd31`

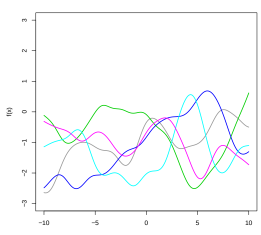
STA571/CS590.01 Due: 21 March, 2014

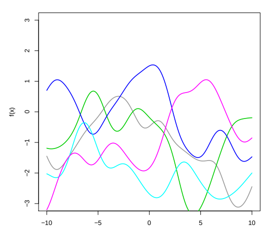


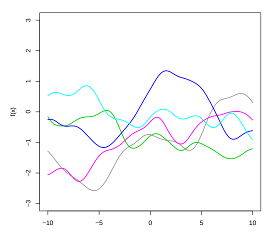


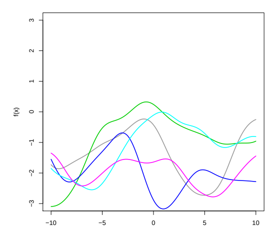


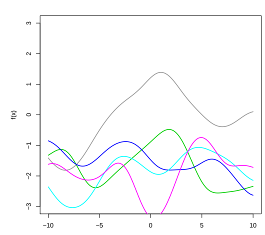






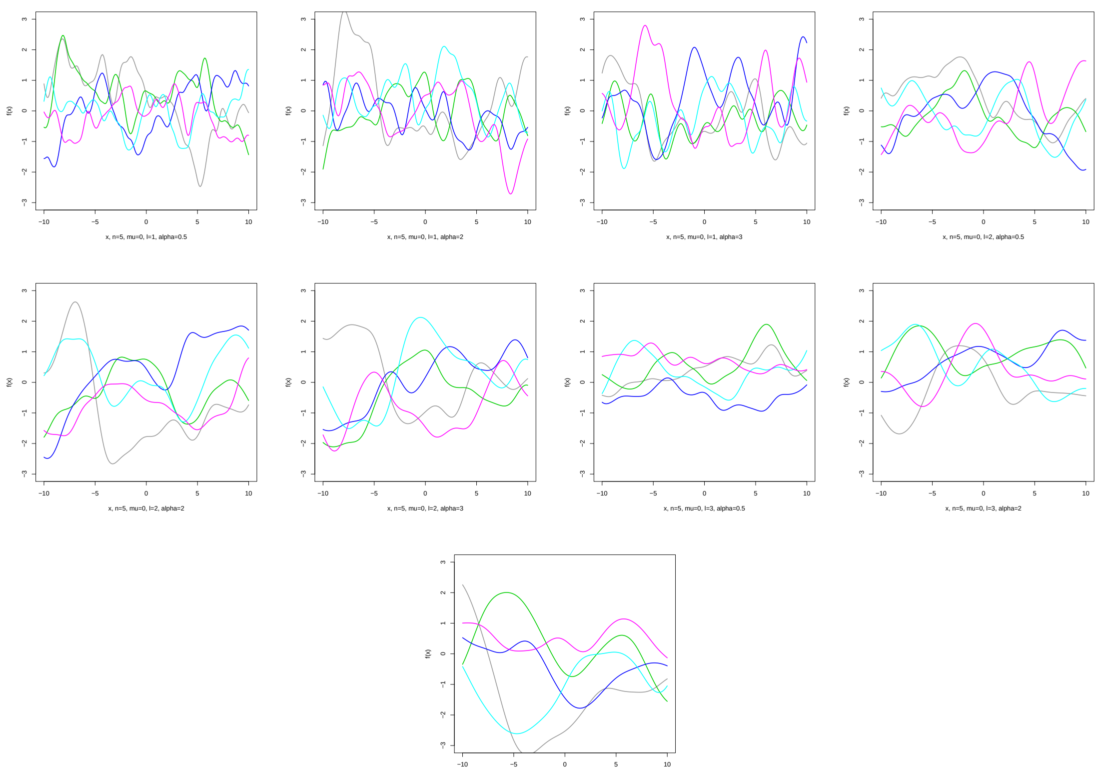






Gaussian Process Simulations

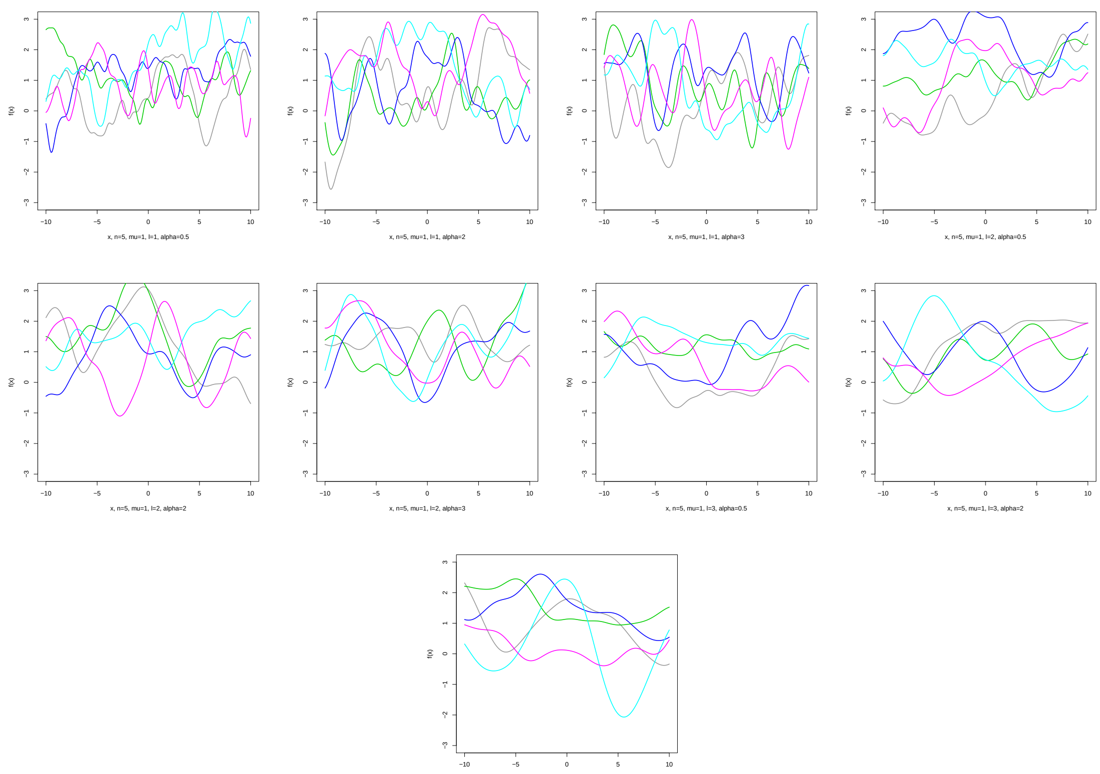
Matt Dickenson `mcd31`
STA571/CS590.01 — Due: 21 March, 2014



Gaussian Process Simulations

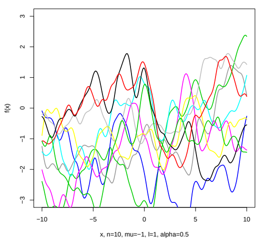
Matt Dickenson `mcd31`

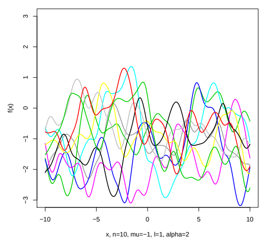
STA571/CS590.01 Due: 21 March, 2014

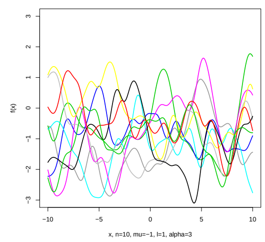


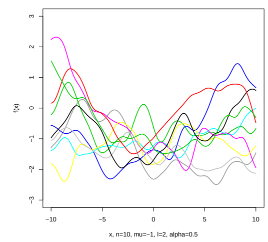
Gaussian Process Simulations

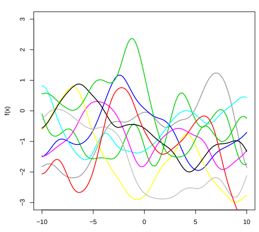
Matt Dickenson `mcd31`
STA571/CS590.01 Due: 21 March, 2014

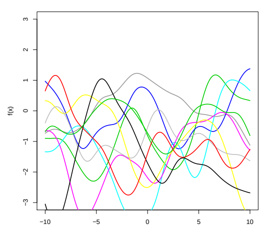


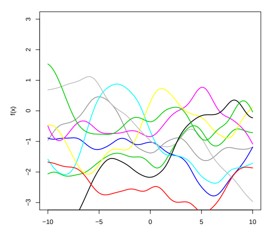


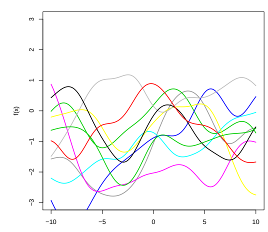


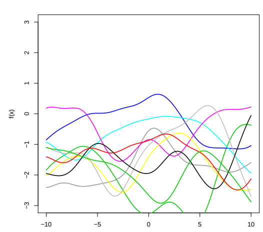






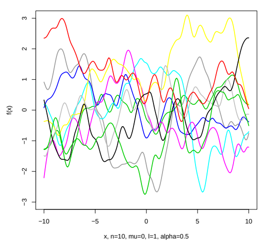


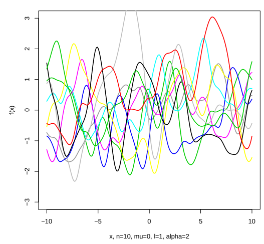


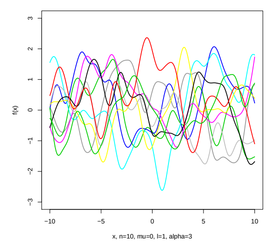


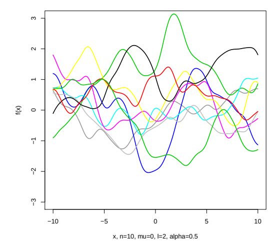
Gaussian Process Simulations

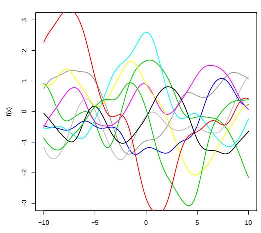
Matt Dickenson `mcd31`
STA571/CS590.01 Due: 21 March, 2014

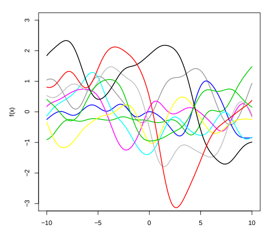


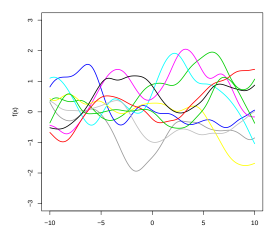


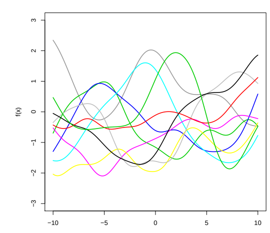


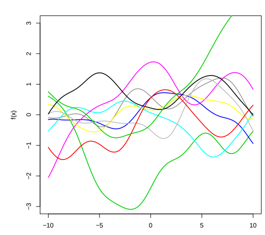






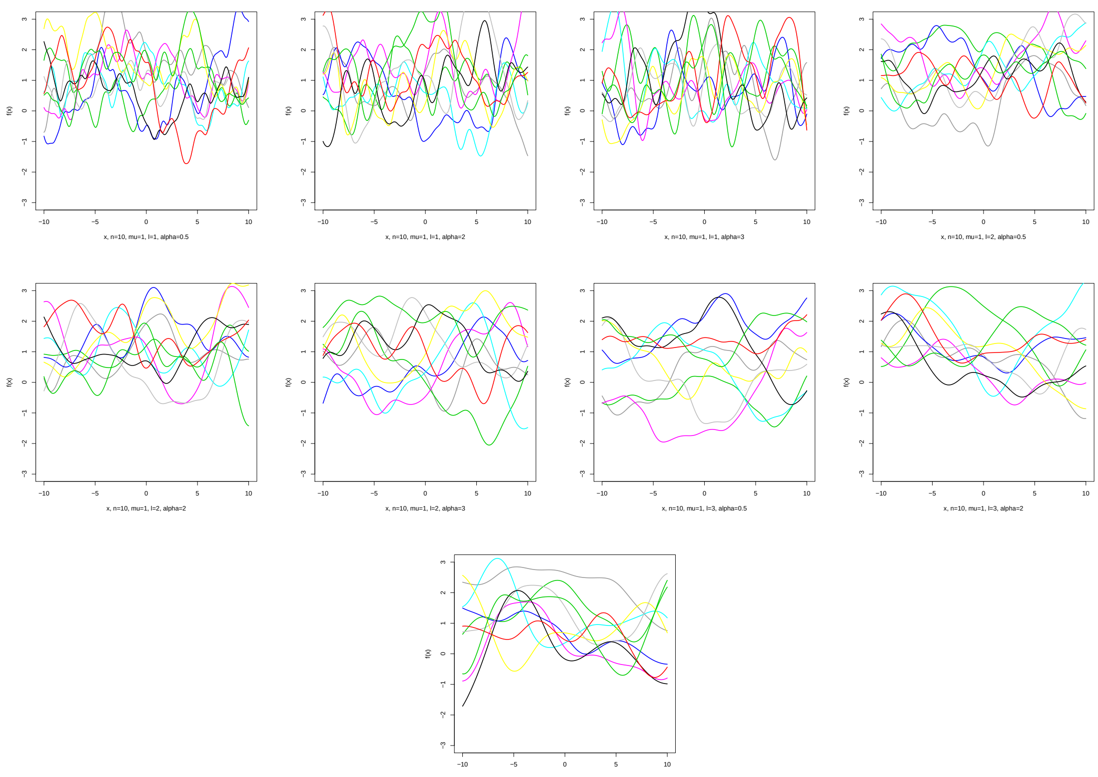


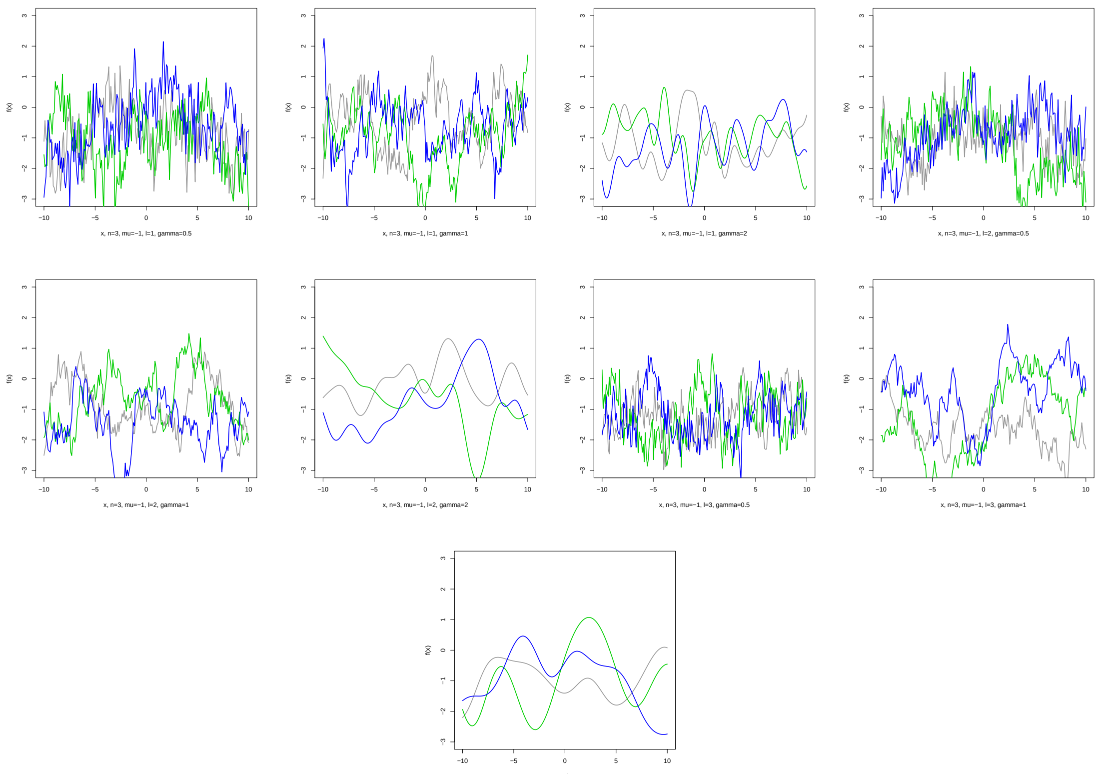




Gaussian Process Simulations

Matt Dickenson `mcd31`
STA571/CS590.01 Due: 21 March, 2014





4 γ-Exponential Covariance

The simulations in this section use the γ-exponential covariance function:

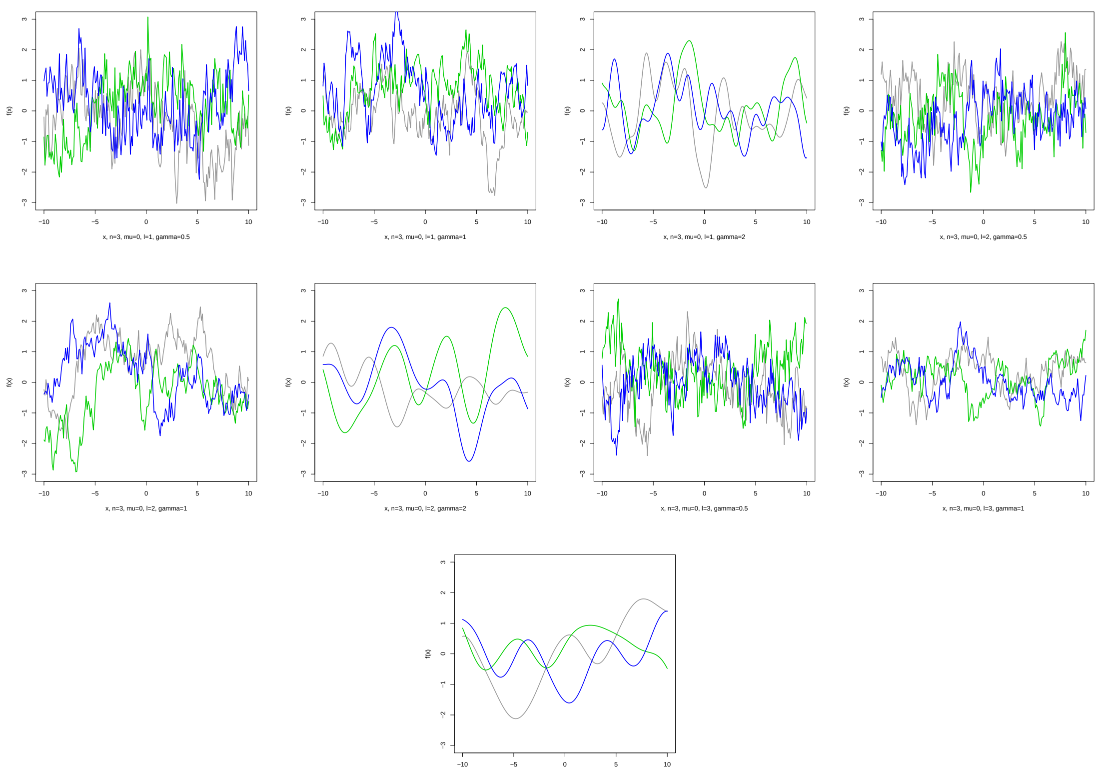
$$k(x, x') = \exp\{-(\frac{x - x'}{\ell})^{\gamma}\}$$

The parameters of interest here are the same as the squared exponential, with the addition of γ. This is a more general class of covariance function, of which the exponential and squared exponential are special cases.

Gaussian Process Simulations

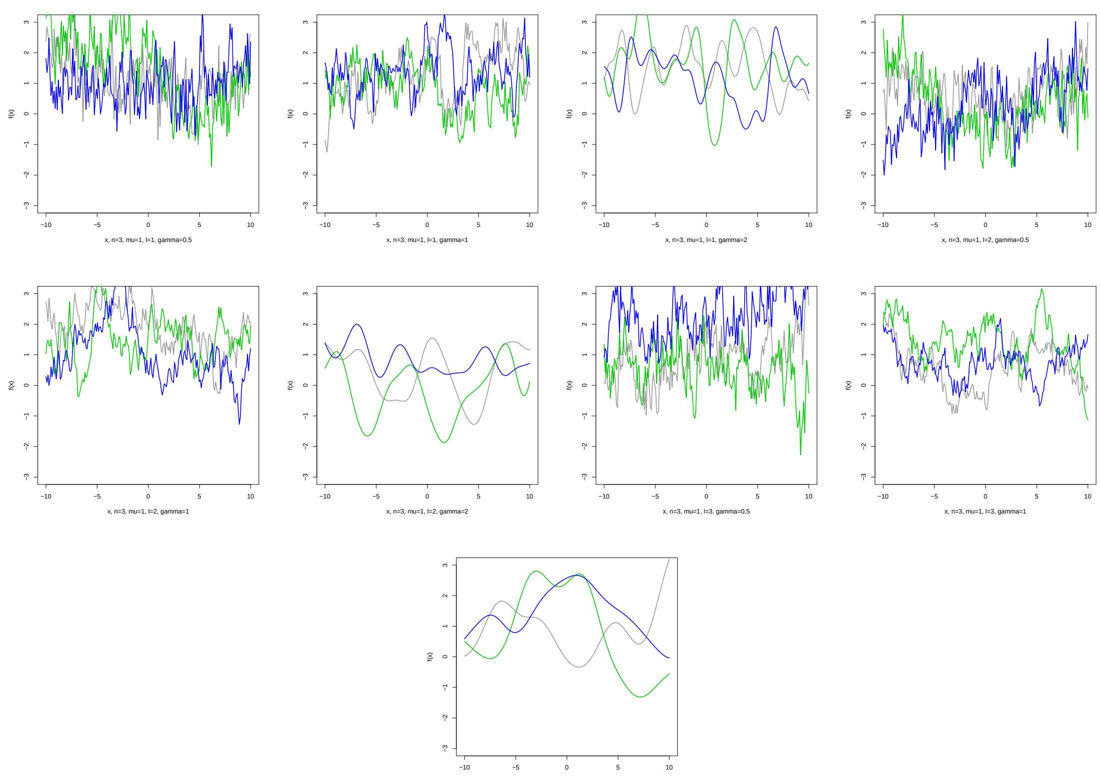
Matt Dickenson `mcd31`

STA571/CS590.01 Due: 21 March, 2014



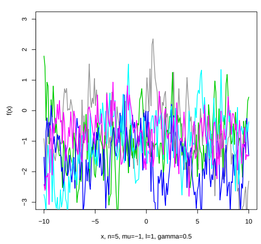
Gaussian Process Simulations

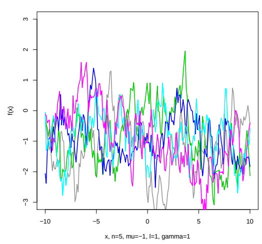
Matt Dickenson `mcd31`
STA571/CS590.01 Due: 21 March, 2014

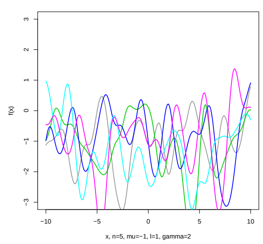


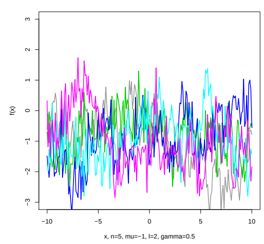
Matt Dickenson mcd31

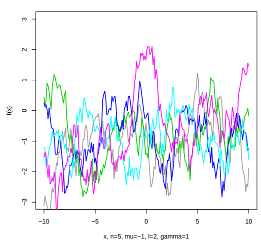
STA571/CS590.01 Due: 21 March, 2014

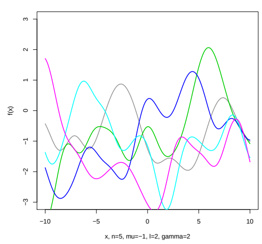


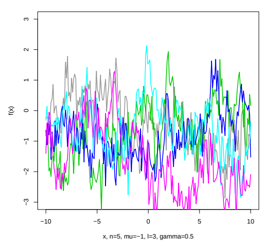


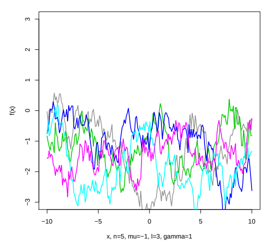


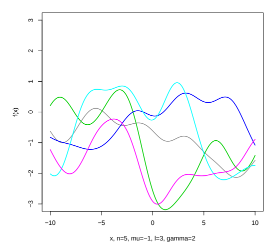








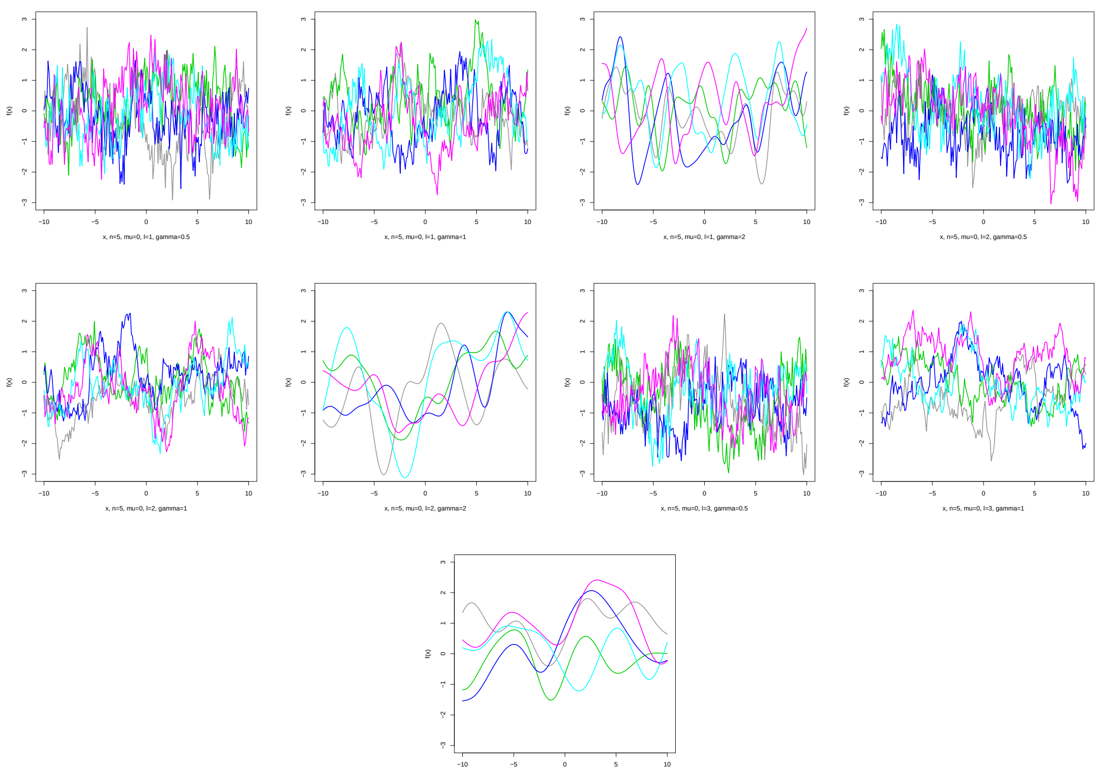




Gaussian Process Simulations

Matt Dickenson `mcd31`

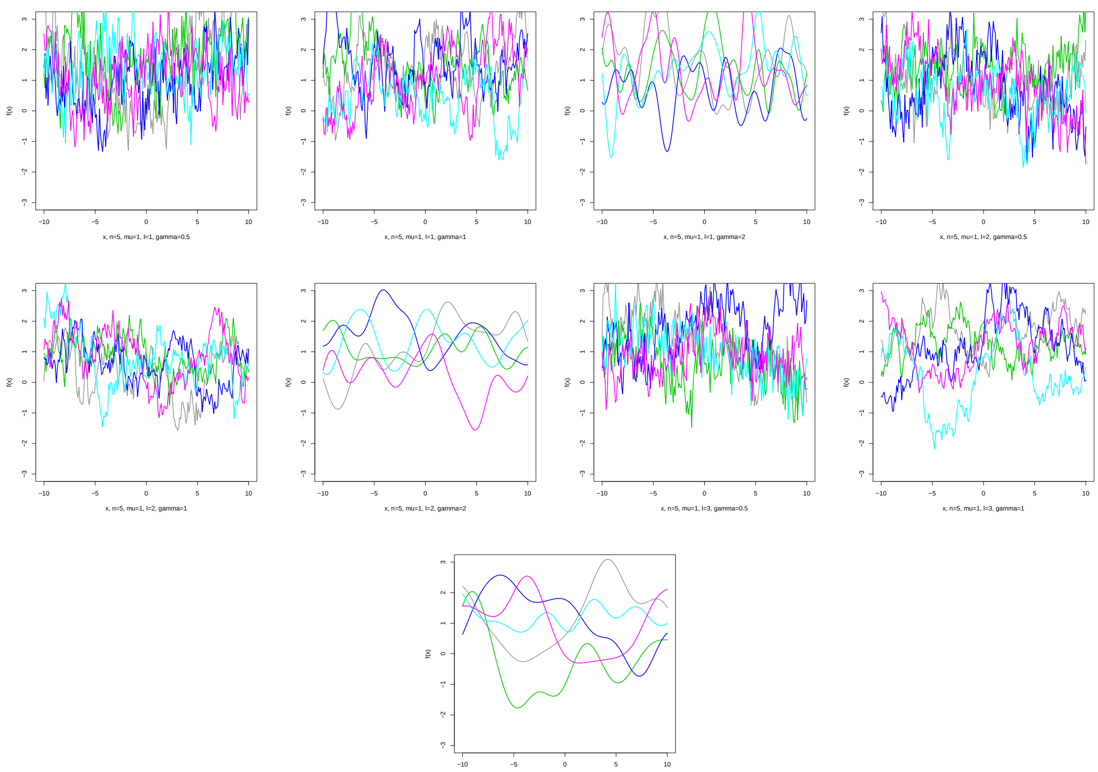
STA571/CS590.01 Due: 21 March, 2014



Gaussian Process Simulations

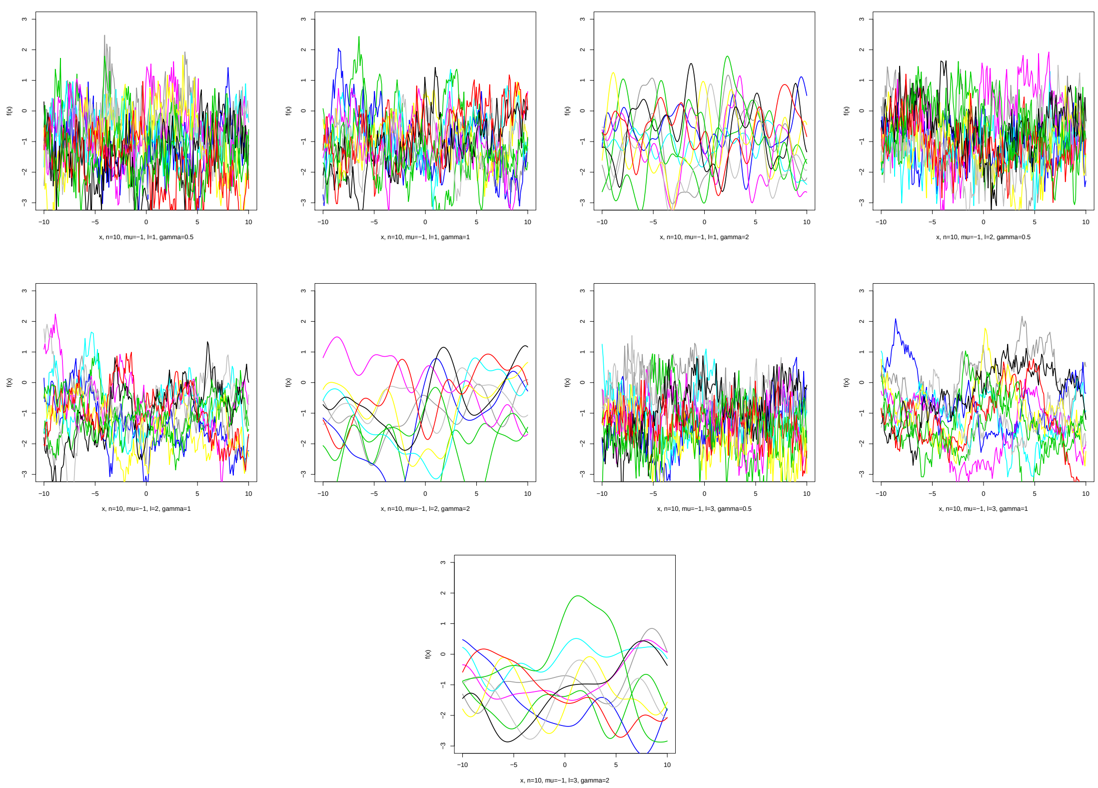
Matt Dickenson `mcd31`

STA571/CS590.01 Due: 21 March, 2014



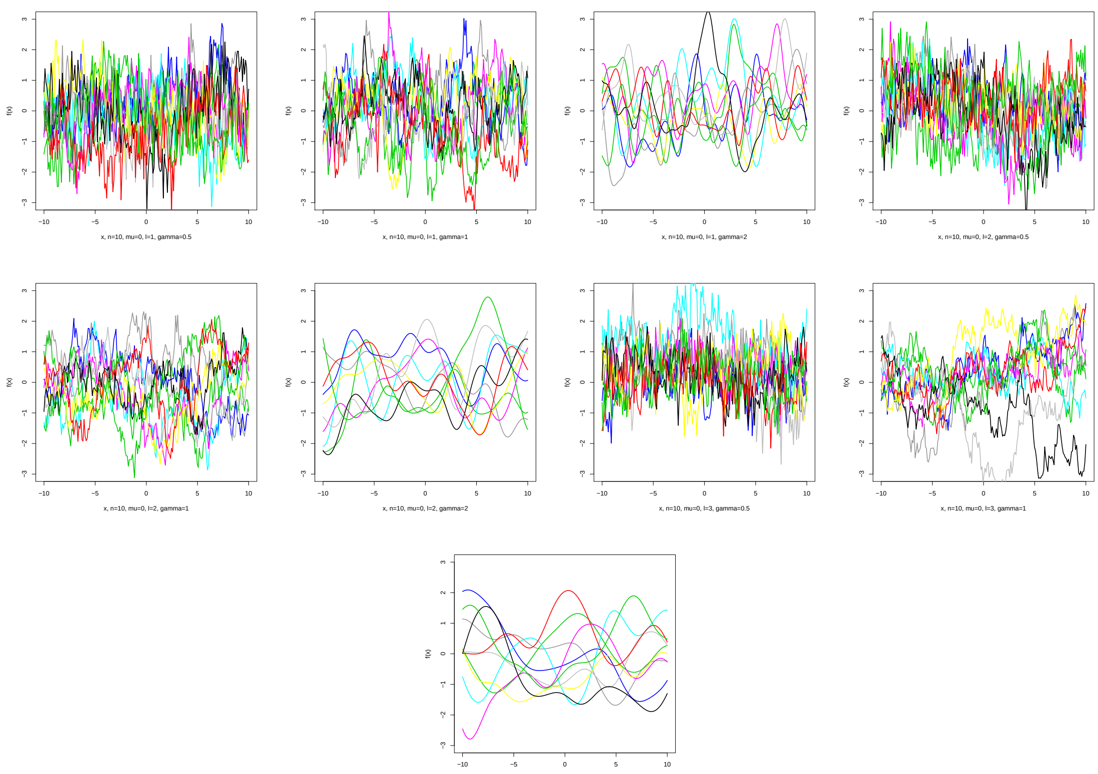
Gaussian Process Simulations

Matt Dickenson `mcd31`
STA571/CS590.01 Due: 21 March, 2014



Gaussian Process Simulations

Matt Dickenson mcd31
STA571/CS590.01 Due: 21 March, 2014



Gaussian Process Simulations

Matt Dickenson `mcd31`
STA571/CS590.01 Due: 21 March, 2014