PS1QM. Alena Sokolyanskaya

September 28, 2018

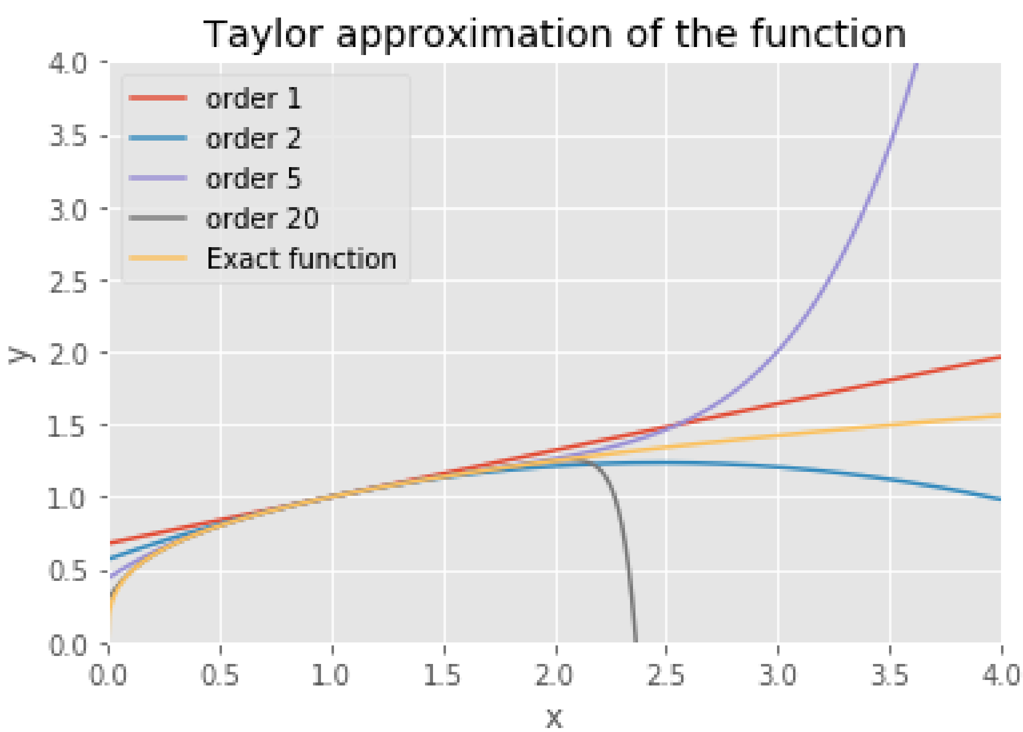
1 Problem Set 1. Quantative Macro

1.1 Question 1. Function Approximation: Univariate

1.1.1 Exercise 1

Approximate $f(x) = x{\cdot}321$ with a Taylor series around $\bar{x} = 1$. Compare your approximation over the domain (0,4). Compare when you use up to 1, 2, 5 and 20 order approximations. Discuss your results.

```
Taylor expansion at n=1 0.321*x + 0.679
Taylor expansion at n=2 0.321*x - 0.1089795*(x - 1)**2 + 0.679
Taylor expansion at n=5 0.321*x + 0.0300570779907967*(x - 1)**5 - 0.040849521596625*(x - 1)**4 +
Taylor expansion at n=20 0.321*x - 0.00465389246518441*(x - 1)**20 + 0.00498302100239243*(x - 1)
```

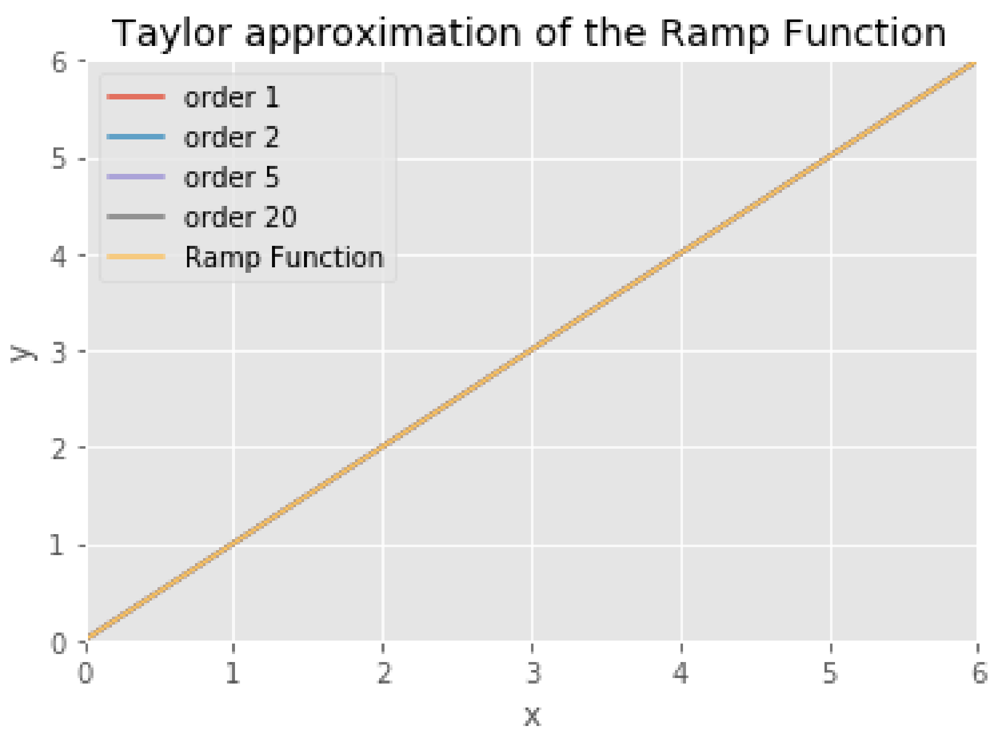


The approximation error, basically the gap between the exact function and approximations, reaches its minimum around $x = 1$ - singularity point. It increases significantly for $x > 2$, and the higher the order of Taylor approximation the higher the error. Taylor series approximations do not distribute the approximation error fairly. The error is small at $x = x_0$, but its error tends to be much larger at the edges of its range.

1.1.2 Exercise 2

Approximate the ramp function $f(x) = \frac{x+|x|}{2}$ with a Taylor series around $\bar{x} = 2$. Compare your approximation over the domain (0,6). Compare when you use up to 1, 2, 5 and 20 order approximations. Discuss your results.

```
Taylor expansion at n=1 1.0*x
Taylor expansion at n=2 1.0*x
Taylor expansion at n=5 1.0*x
Taylor expansion at n=20 1.0*x
```



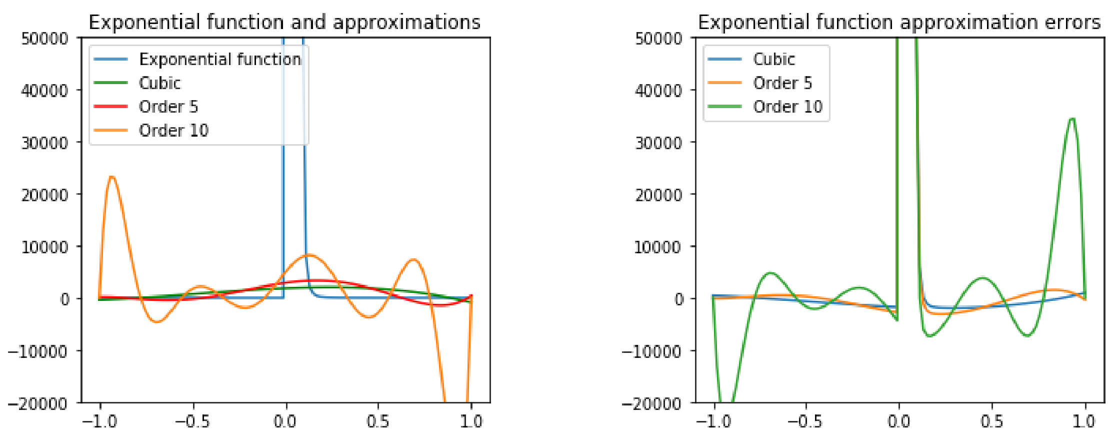
The ramp function is a unary real function, whose graph is shaped like a ramp. The approximation fits perfectly the exact function. However, since the ramp function takes 0 for negative inputs, approximation fails for the values lower than 0.

1.1.3 Exercise 3

Approximate these three functions: $e^{\frac{1}{x}}$, the runge function $\frac{1}{1+25x^2}$, and the ramp function $f(x) = \frac{x+|x|}{2} \$ for the domain x \in [-1, 1]$ with:

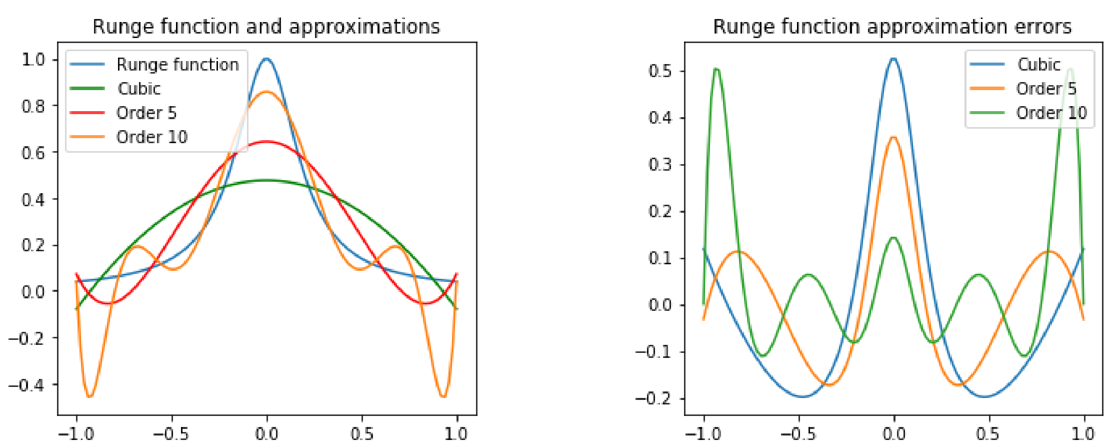
3.1. Evenly spaced interpolation nodes and a cubic polynomial

(a) Exponential function



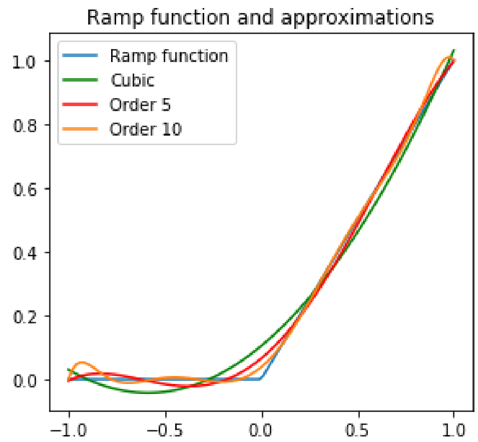
For the approximation of the exponential function in case of evenly spaced nodes we can see that none of the approximation succeed: the errors are high at the tails and also in the center.

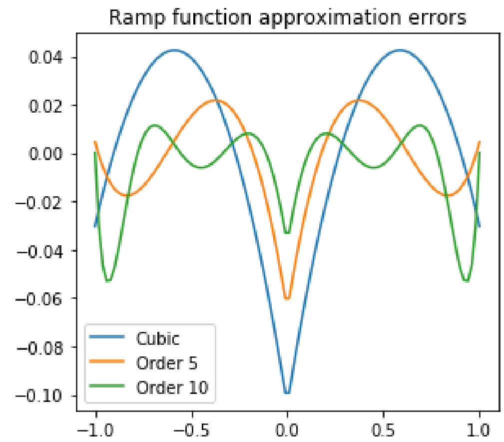
(b) Runge Function



There are smooth functions for which polynomial with evenly spaced nodes rapidly deteriorate: a classic example is the Runge's function where the approximation error rises rapidly with the number of nodes.

(c) Ramp Function

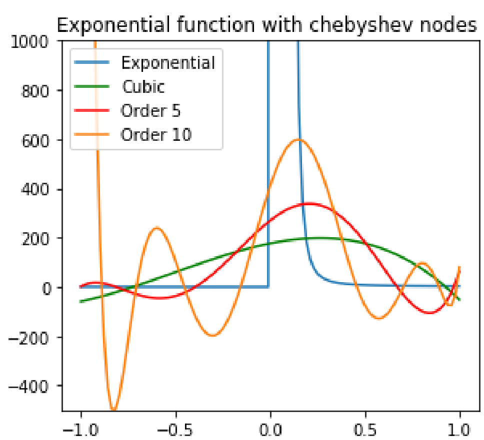


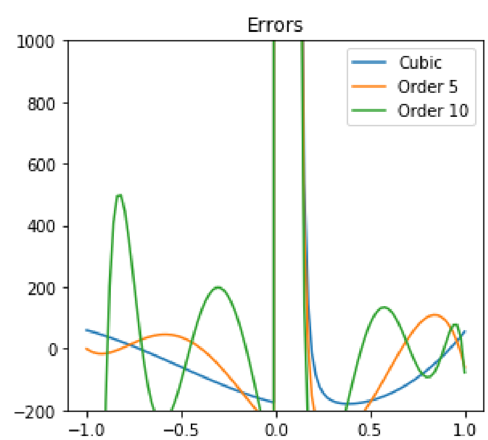


Increasing the order of monomial we see that they fit better than cubic.

3.2. Chebyshev interpolation nodes and a cubic polynomial [.] The Chebyshev nodes are the j roots of the Chebyshev polynomials which are actually defined on the [-1; 1] interval. These nodes are not evenly spaced. They are more closely spaced near the endpoints of the interpolation interval and less so near the center, this avoids more easily instabilities at the endpoints.

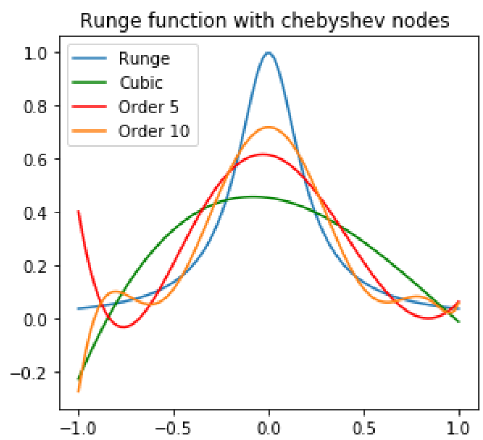
(a) Exponential Function

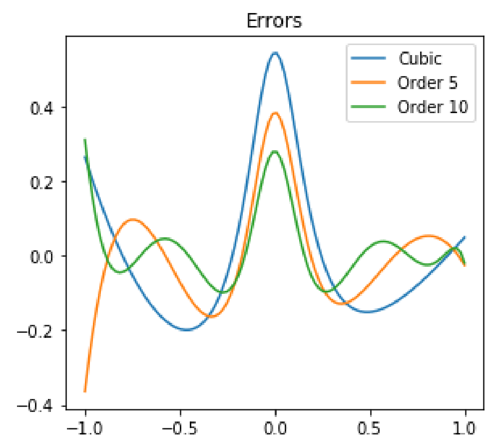




Again for exponential function we see that also using chebyshev interpolation nodes approximations have high errors.

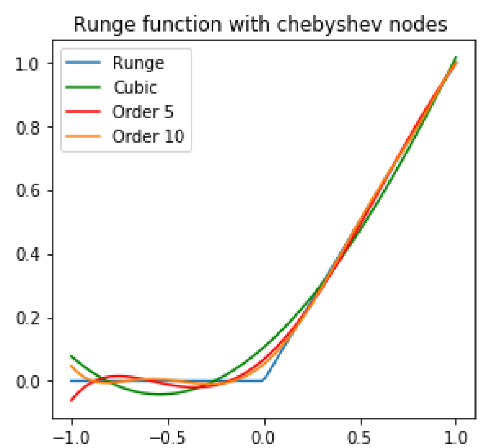
(b) Runge Function





For runge function increasing order of polynomial decreses the errors at the tails. But compared to the previous point where we used evenly spaced nodes, n chebyshev interpolation case errors are lower.

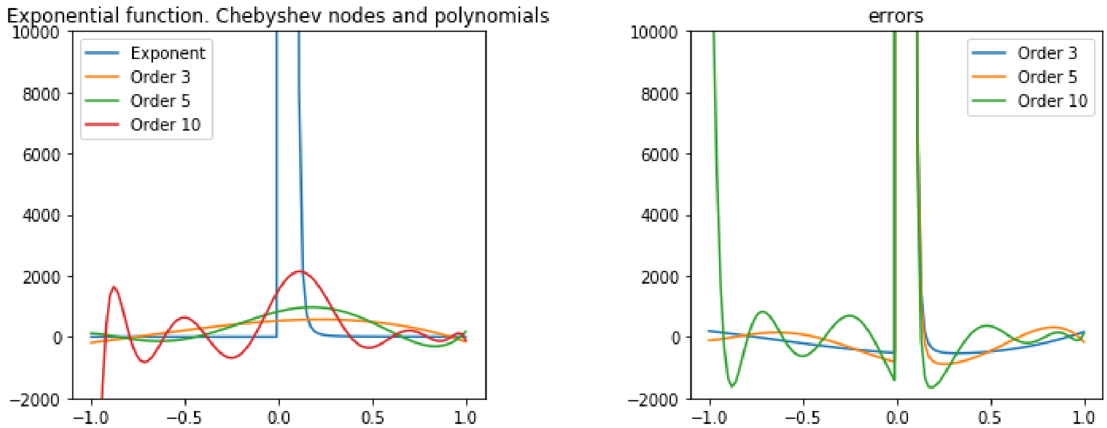
(c) Ramp Function



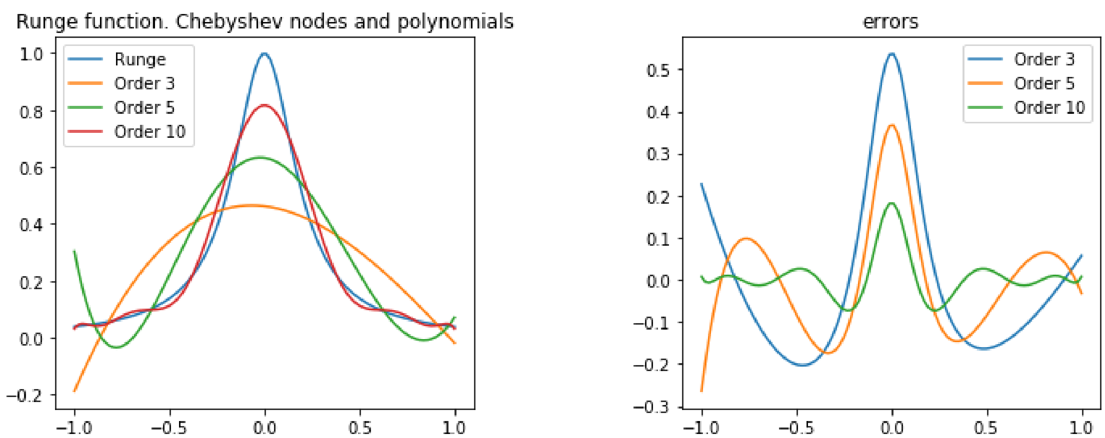
At point x=0 all approximations fail, and similar to runge function incresing order of polynomials lead to dicrease of errors at the tails.

3.3. Chebyshev interpolation nodes and Chebyshev polynomial of order 3, 5 and 10 Results for chebyshev polynomial and cubic polynomial are very similar.

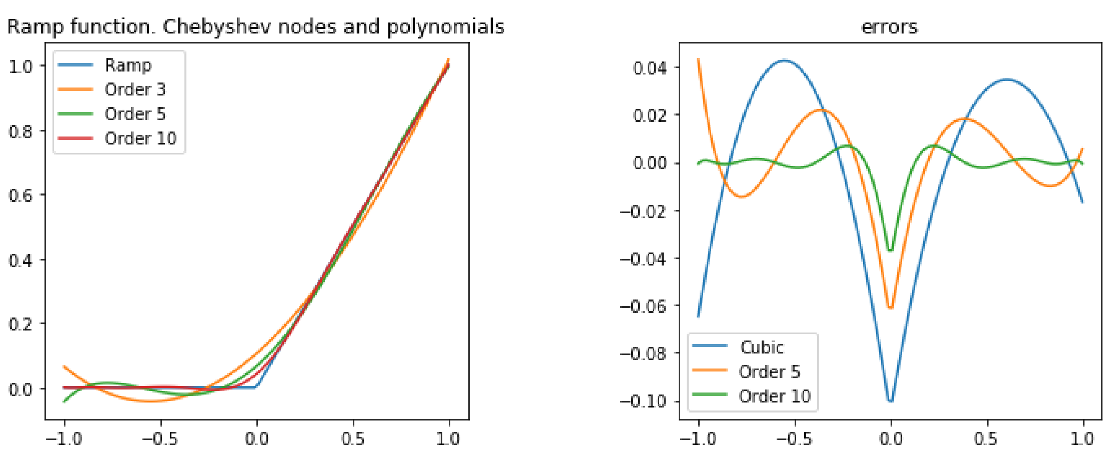
(a) Exponential function



(b) Runge function



(c) Ramp function



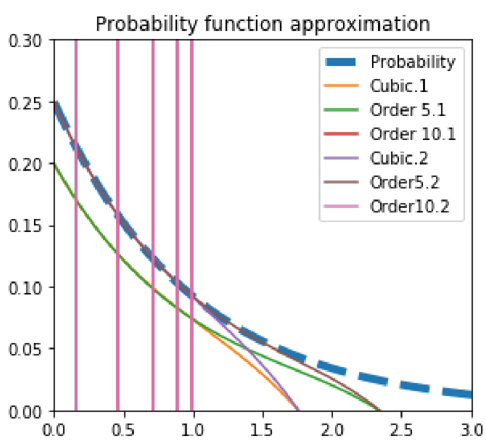
1.1.4 Exercise 4

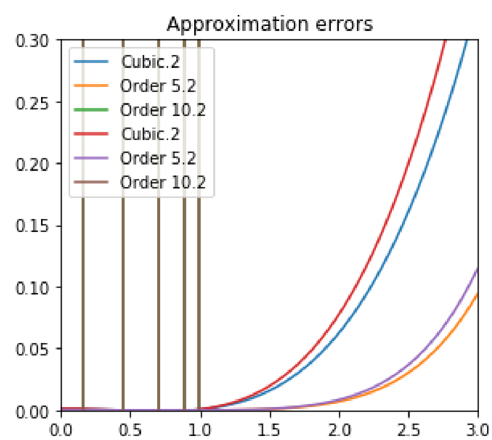
Approximate the following probability function: $p(x) = \dfrac{e^{-\alpha x}}{\rho_1 + \rho_2 e^{-\alpha x}}$ for the domain $x \in (0, 10]$ using Chebyshev interpolation nodes and Chebyshev polynomial of order 3, 5 and 10. Report your approximation and errors. Do this for two combinations of paramters $\alpha = 1.0$, $\rho_2 = \dfrac{1}{100}$ and $\rho_1 = \dfrac{1}{0.2}$. Redo for $\rho_1 = \dfrac{1}{0.25}$. Plot the results for both combinations in the same graph.

Case 1 for $\rho_2 = \dfrac{1}{100}$

Case 2 for $\rho_1 = \dfrac{1}{0.25}$

Again increasing order of the polynomial leads to lower approximation errors.





1.1.5 Question 2. Function Approximation: Multivariate

Consider the following CES function $f(k,h) = \left[(1-\alpha)k^{\frac{\sigma-1}{\sigma}} + \alpha h^{\frac{\sigma-1}{\sigma}}\right]^{\frac{\sigma}{\sigma-1}}$ where σ) is the elasticity of substitution (ES) between capital and labor and α is a relative input share parameter. Set $\alpha = 0.5$, $\sigma = 0.25$, $k \in [0, 10]$ and $h \in [0, 10]$. Do the following items:

(1) Show that σ is the ES (analytically). Given the function $f(k,h) = \left[(1-\alpha)k^{\frac{\sigma-1}{\sigma}} + \alpha h^{\frac{\sigma-1}{\sigma}}\right]^{\frac{\sigma}{\sigma-1}}$ let $\dfrac{\sigma-1}{\sigma} \equiv \rho$

Then our function becomes: $f(k,h) = [(1-\alpha)k^\rho + \alpha h^\rho]^{\frac{1}{\rho}}$

The MPL and MPK are respectively:

$$f_h(k,h) = \frac{\delta f}{\delta h} = \frac{1}{\rho}((1-\alpha)k^\rho + \alpha h^\rho)^{\frac{1}{\rho}-1} \alpha\rho h^{\rho-1}$$

$$f_k(k,h) = \frac{\delta f}{\delta k} = \frac{1}{\rho}((1-\alpha)k^\rho + \alpha h^\rho)^{\frac{1}{\rho}-1} (1-\alpha)\rho k^{\rho-1}$$

Where f is a differentiable real-valued function of a single variable, we define the elasticity of $f(x)$ with respect to x (at the point x) to be: $\sigma(x) = \frac{xf'(x)}{f(x)} \equiv \frac{\delta d(x)/f(x)}{\delta x/x}$

Let $\frac{\delta ln(f(x))}{\delta ln(x)} = \frac{\delta v}{\delta u}$. By chain rule we get $\frac{\delta v}{\delta u} = \frac{\delta v}{\delta x}\frac{\delta x}{\delta u} = \frac{f'(x)}{f(x)}x$

Therefore,

$$ln(\frac{f_k}{f_h}) = ln\frac{\frac{\delta f}{\delta h} = \frac{1}{\rho}((1-\alpha)k^\rho + \alpha h^\rho)^{\frac{1}{\rho}-1} \alpha\rho h^{\rho-1}}{\frac{\delta f}{\delta k} = \frac{1}{\rho}((1-\alpha)k^\rho + \alpha h^\rho)^{\frac{1}{\rho}-1} (1-\alpha)\rho k^{\rho-1}} = ln(\frac{\alpha}{1-\alpha}(\frac{h}{k})^{\rho-1}) = \frac{\alpha}{1-\alpha}(1-\rho)ln(\frac{k}{h})$$

We get the following: $ln(\frac{k}{h}) = \frac{1}{1-\rho}\frac{1-\alpha}{\alpha}ln(\frac{f_k}{f_h})$

The term $\frac{1}{1-\rho}\frac{1-\alpha}{\alpha}$ is the elasticity of substitution. Since we know that $\alpha = 0.5$ and substituting $\frac{\sigma-1}{\sigma} \equiv \rho$ we get that substitution of elasticity is equal to $\frac{1}{1-\frac{\sigma-1}{\sigma}} = \sigma$

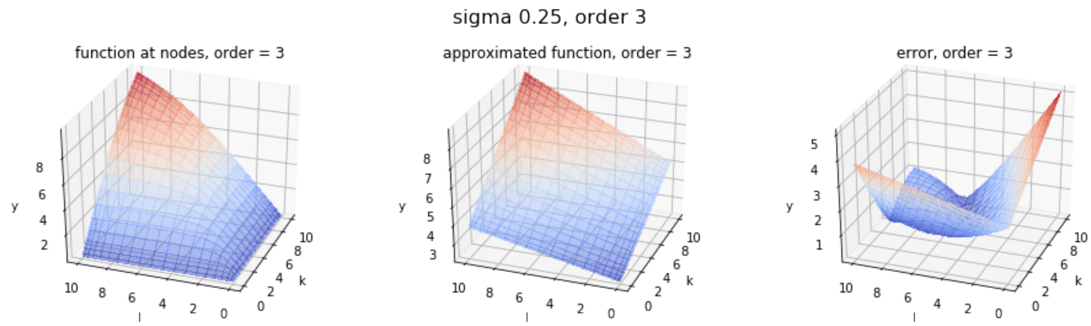
(2) Compute labor share for an economy with that CES production function assuming factor inputs face competitive markets (analytically) If markets are competitive $f_h = \alpha h^{-\frac{1}{\sigma}} f(k,h)^{-\frac{1}{\sigma}}$

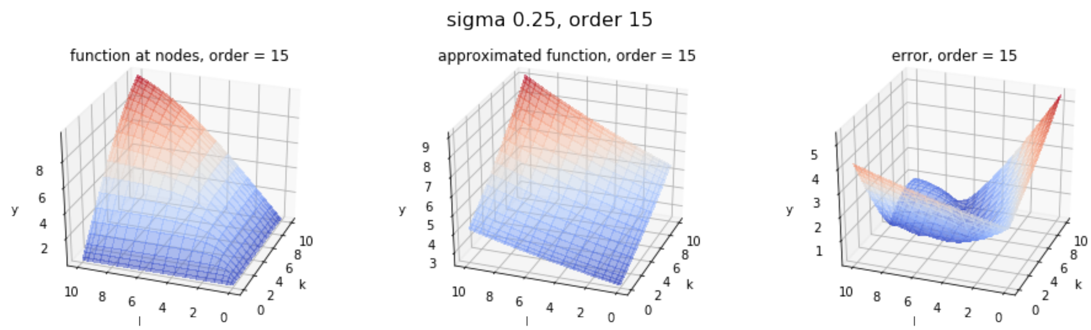
Labor share is equal to $f_h\frac{h}{f(k,h)} = \alpha(\frac{h}{f(k,h)})^{1-\frac{1}{\sigma}}$

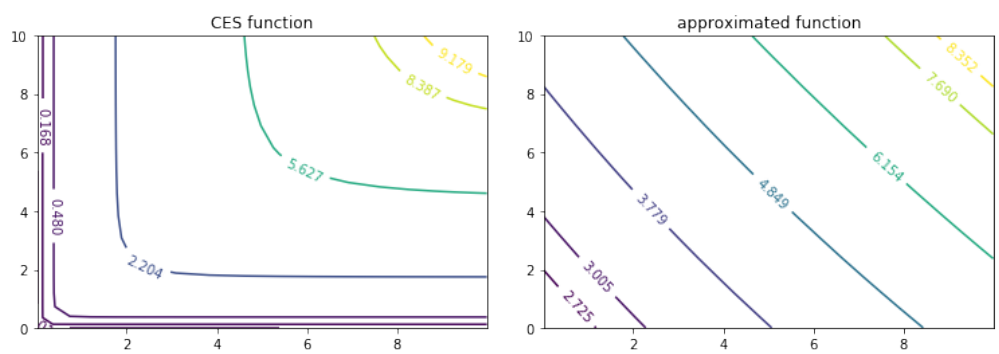
Approximate $f(k,h)$ using a 2-dimensional Chebyshev regression algorithm. Fix the number of nodes to be 20 and try Cheby polynomials that go from degree 3 to 15. For each case, plot the exact function and the approximation (vertical axis) in the (k, h) space CES function $f(k,h) = \left[(1-\alpha)k^{\frac{\sigma-1}{\sigma}} + \alpha h^{\frac{\sigma-1}{\sigma}}\right]^{\frac{\sigma}{\sigma-1}}$ where σ is the elasticity of substitution (ES) between capital and labor and α is a relative input share parameter. Set $\alpha = 0.5$, $\sigma = 0.25$, $k \in [0,10]$ and $h \in [0,10]$.

Steps of Chebyshev Approximation Algorithm in R^2 Step 1: Compute the $m \geq n+1$ Chebyshev interpolation nodes on $[-1,1]$ Step 2: Adjust nodes to $[a,b]$ and $[c,d]$ intervals Step 3: Evaluate f at approximation nodes Step 4: Compute Chebyshev coefficients

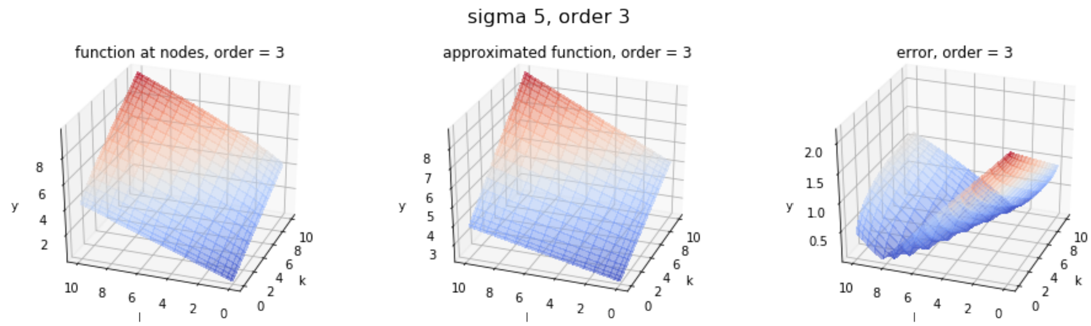
For all the cases with higher degree polynomial the errors slightly decrease.
CASE 1. $\sigma = 0.25$

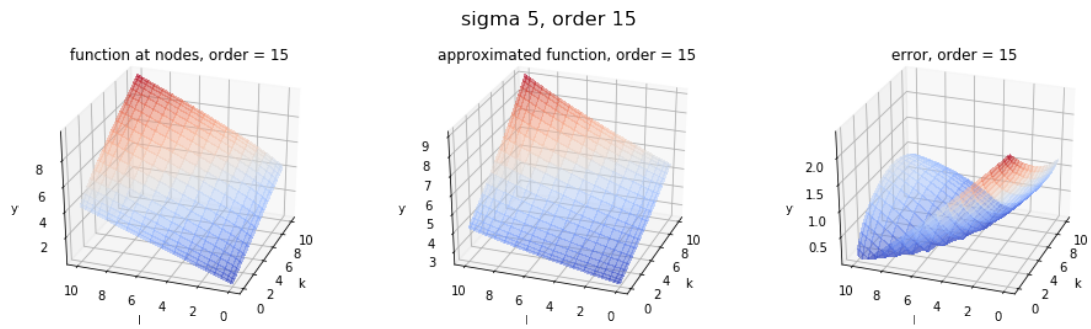


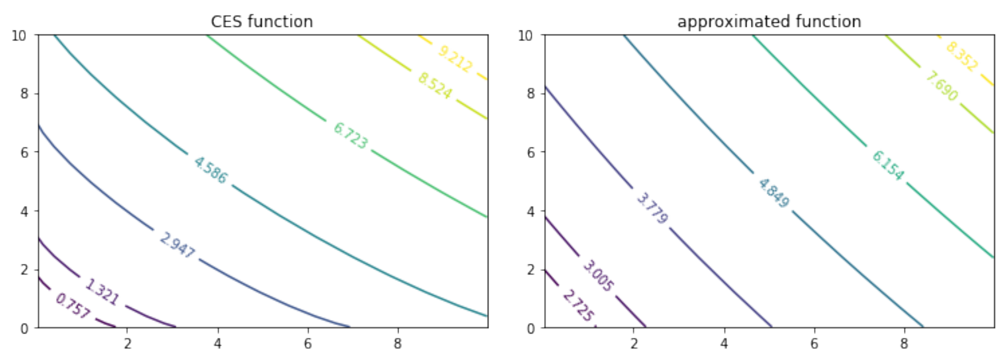




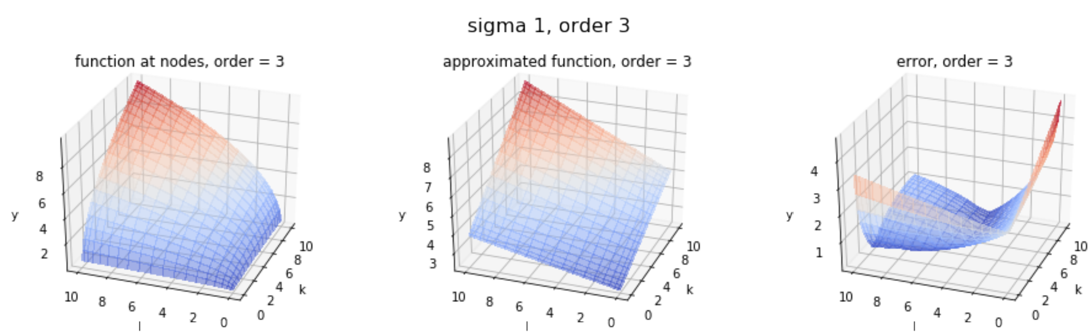
CASE 2. $\sigma = 5$

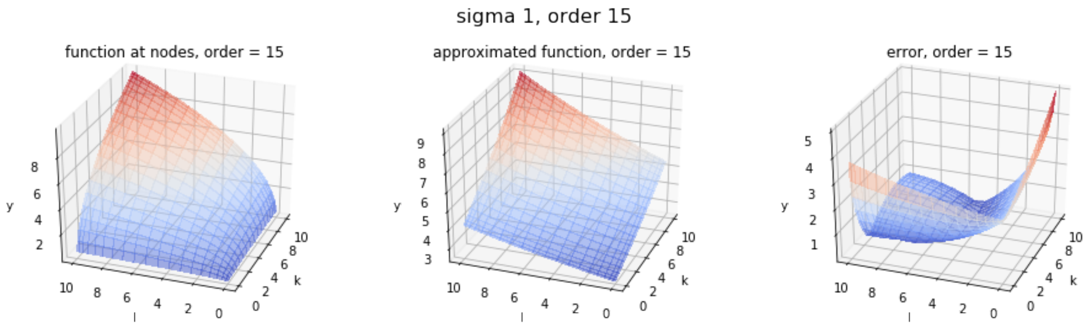


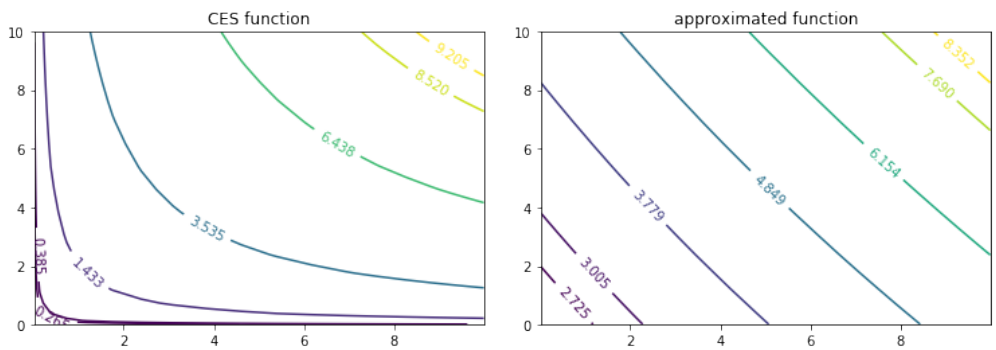




CASE 3. $\sigma = 1$ in this case we adjust $\sigma = 0.999999$




sigma 1, order 15
function at nodes, order = 15
approximated function, order = 15
error, order = 15
y
k
l


CES function
approximated function
9.205
8.520
6.438
3.535
1.433
0.385
8.352
7.690
6.154
4.849
3.779
3.005
2.725