PS 7

Due: Mon, Apr 13

1: By the Book Section 9.4, Problem 6. For part (b), it may help to refer to a homework problem earlier in the semester!

Answer: For the first part: Newton solves a linear approximation, and in this case the original equation is linear. Therefore, Newton converges in one step.

For the second part, consider convergence to the multiple root of $x^3 = 0$, which is linear. We did this example early in the semester.

2: Naive Newton Consider the nonlinear system of equations

$$\begin{aligned} x^2 + xy^2 &= 9 \\ 3x^2y - y^3 &= 4 \end{aligned}$$

Fill in the following Newton iteration code:

```
for k = 1:20
  F = % Your code here
  if norm(F) < rtol
    break;
  end
  J = % Your code here
  dx = J\F;
  x = x-dx;
end
```

Run your code with an initial guess of $(1, 1)$ and a residual norm tolerance of 10^{-12}. Do you see quadratic convergence?

Answer: The code follows:

```
x = [1;  1];
for k = 1:20
  F = [ x(1)^2 + x(1)*x(2)^2 - 9;
        3*x(1)^2*x(2) - x(2)^3 - 4 ];
  fprintf('%d: %e\n', k, norm(F));
  if norm(F) < 1e-12
    break;
  end
```

```
  J = [ 2*x(1) + x(2)^2,  2*x(1)*x(2);
        6*x(1)*x(2),      3*x(1)^2 - 3*x(2)^2];
  dx = J\F;
  x = x-dx;
end
```

The residual was

```
1: 7.280110e+00
2: 4.875347e+01
3: 1.395474e+01
4: 2.916736e+00
5: 1.355139e-01
6: 6.031975e-05
7: 1.058961e-10
8: 5.024296e-15
```

From steps 5–8, convergence is clearly quadratic (with roundoff dominating by step 8); that is, $e_6 \approx e_5^2$ and $e_7 \approx e_6^2$.

3: Continue with Care Consider the boundary value problem

$$\begin{aligned} v''(x) + \gamma \exp(v(x)) &= 0, \quad 0 < t < 1 \\ v(0) = v(1) &= 0 \end{aligned}$$

discretized via finite differences on a mesh with 100 equally spaced points; see example 9.3 in the book.

- Write a code to find v for a range of γ values from 1 to 3.5 (use `gammas = linspace(1,3.5)` to generate the mesh). For the first value of γ, you should use an initial guess of $v = 0$; for subsequent values, use the value of γ at the previous iterate. Plot all your solutions together on a single plot.
- For all γ in the given range, the Jacobian matrix at the solution remains negative definite. Plot $\lambda_{\max}(J(x^*))$ (the eigenvalue closest to zero) as a function of γ. What do you notice?
- Try running your code again, this time going up to a maximum value of 4 rather than 3.5. What happens?

Note: You may start from the following code

```
n = 100;
h = 1/(n+1);
T = diag(ones(n−1,1),−1) + diag(one(n−1,1),1) − 2*eye(n);
v = zeros(n,1);
```

If n was very large, we might want to use a sparse matrix[1], but it's probably not worth it in this case.

Answer: My code follows:

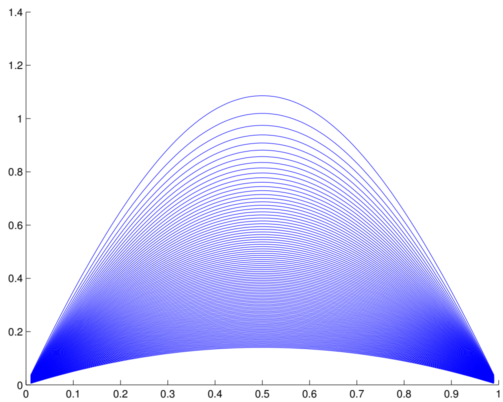
```
n = 100;
h = 1/(n+1);
T = diag(ones(n−1,1),−1) + diag(ones(n−1,1),1) − 2*eye(n);
v = zeros(n,1);
x = linspace(0,1,n+2);
x = x(2:end−1);
gammas = linspace(1,3.5);
lambdas = [];

figure(1);
hold on
for gamma = gammas
  for k = 1:10
    F = h^−2 * (T*v) + gamma * exp(v);
    J = h^−2 * T + diag(gamma * exp(v));
    v = v−J\F;
  end
  lmax = max(eig(J));
  lambdas = [lambdas, lmax];
  plot(x, v)
end

figure(2);
plot(lambdas)
```

It would probably be smarter to put a convergence criterion on the loop, but this is fast enough that I was willing to be lazy. The code blows up if

[1] I'd probably switch to a more accurate discretization method, first, but that's a topic for CS 4210.



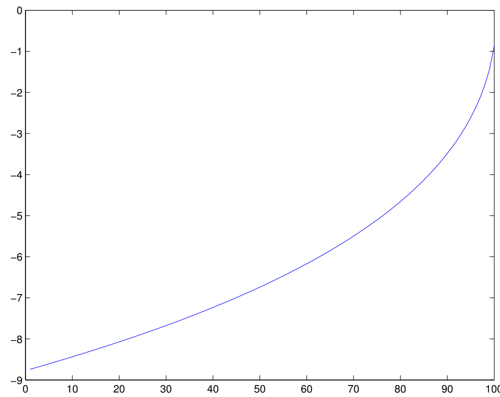


Figure 1: Solutions to $v'' + \gamma \exp(v) = 0$ for $\gamma \in [1, 3.5]$ (left) and maximum eigenvalue for the discrete Jacobian vs γ (right). For γ even a little larger than 3.5, the Jacobian matrix becomes singular at a turning point (and the solution cannot be continued to larger γ values).

the upper bound is much greater than 3.5. The plots show that the Jacobian likely becomes singular for γ a bit above 3.5; that γ represents a turning point for the solution manifold, and the solution branch does not exist for larger γ. This shows up when actually trying to run the computation because the Newton iteration starts to diverge, despite the continuation strategy.