

# sh\_transform-Copy1

January 30, 2022

## 0.0.1 Solving Poisson Equation on $S^2$ : $\Delta u = f$

```
[1]: import my_backends
      from my_backends.ducc0_wrapper import *
      import my_shcoeffs
      from my_shcoeffs import SHCoeffs
      ## my_shcoeffs contains expand_adjoint_analysis
```

```
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plmbar.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plmbar_d1.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plbar.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plbar_d1.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plmon.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plmon_d1.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plon.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plon_d1.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plmschmidt.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plmschmidt_d1.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plschmidt.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plschmidt_d1.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plegendrea.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plegendrea_d1.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPIIN/doc/plegendre.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
```

```

reu-2021/PDPINN/doc/plegendre_d1.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shexpanddh.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/makegriddh.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shexpanddhc.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/makegriddhc.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shglq.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shexpandglq.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/makegridglq.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shexpandglqc.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/makegridglqc.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/glqgridcoord.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shexpandlsq.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shexpandwlsq.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/makegrid2d.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/makegridpoint.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/makegridpointc.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shmultiply.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shread2.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shread2error.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shreadjpl.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shreadjplerror.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shcilmtovector.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shvectortocilm.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shcilmtoindex.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-

```

reu-2021/PDPINN/doc/shcindextocilm.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shrtoc.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shctor.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shadmitcorr.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shconfidence.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shmultitaperse.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shmultitapercse.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shlocalizedadmitcorr.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shreturntapers.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shreturntapersm.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/computedm.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/computedg82.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shfindlwin.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shbiask.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shmtcouplingmatrix.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shbiasadmitcorr.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shmtdebias.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shmtvaropt.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shsjkpg.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shmultitapermaskse.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shmultitapermaskcse.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shreturntapersmap.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shbiaskmask.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/computedmap.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-

reu-2021/PDPINN/doc/curve2mask.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shbias.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/sphericalcapcoef.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/djpi2.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shrotatecoef.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/shrotaterealcoef.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/makegravgriddh.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/makegravgradgriddh.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/makegeoidgriddh.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/cilmlplusdh.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/cilmlminusdh.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/cilmlplusrhohdh.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/cilmlminusrhohdh.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/batohilmdh.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/batohilmrhohdh.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/downcontfilterma.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/downcontfiltermc.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/normalgravity.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/makemaggriddh.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/makecirclecoord.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/makeellipsecoord.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/wigner3j.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/dhaj.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-reu-2021/PDPINN/doc/makemaggradgriddh.doc'

[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-

```

reu-2021/PDPINN/doc/shrotatetapers.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/slepiancoeffs.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/slepiancoeffstosh.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shscouplingmatrix.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shmtvar.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shslepianvar.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/shscouplingmatrixcap.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/makegravgridpoint.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/makemaggridpoint.doc'
[Errno 2] No such file or directory: '/Users/aparnagupte/Documents/cornell-
reu-2021/PDPINN/doc/makegradientdh.doc'

```

```

[2]: import torch

N = 20
from Basis import SPHBasis
from Problem import Problem_Sphere_Poisson
from Model import Model
from Net import Sphere_Net, SPH_Sphere_Net
import seaborn as sns
import matplotlib.pyplot as plt
import math
from math import pi
from spherical_harmonics import *
from torch.distributions.uniform import Uniform
from numpy import arccos
import numpy as np

import pyshtools
from pyshtools.shio import shread
from pyshtools.expand import MakeGridDH
from pyshtools.expand import SHEexpandDH
from pyshtools.spectralanalysis import spectrum

torch.manual_seed(0)
maxiter = 10000
problem = Problem_Sphere_Poisson()

```

```
[3]: N = 20
azimuth = torch.linspace(0.0001, pi, N, requires_grad=False)
polar = torch.linspace(0, 2*pi, 2*N, requires_grad=False)

azimuth, polar = torch.meshgrid(azimuth, polar)
location = torch.cat([azimuth.reshape(-1, 1), polar.reshape(-1, 1)], dim=1)
location.shape
```

```
[3]: torch.Size([800, 2])
```

```
[4]: def construct_model(net):
    class Poisson1dModel(Model):
        def add_loss_history(self):
#             self.loss_history.append([self.bc_loss, self.pde_loss, self.
→ predict_error_value])
            self.loss_history.append([self.pde_loss, self.predict_error_value])

        def __init__(self):
            super().__init__(problem=problem, net=net, maxiter=maxiter)

        def inner_sample(self, N=N):
            azimuth = torch.linspace(0.0001, pi, N, requires_grad=False)
            polar = torch.linspace(0, 2*pi, 2*N, requires_grad=False)

            azimuth, polar = torch.meshgrid(azimuth, polar)
            location = torch.cat([azimuth.reshape(-1, 1), polar.reshape(-1,
→ 1)], dim=1)
            return location # numpy ndarray with shape (N, 2*N)

        def bc_sample(self):
            return torch.tensor([[1., 1.]]) # arbitrary point is okay?

        def init_sample(self):
            pass

        def plot(self, net):
            azimuth = torch.linspace(0, math.pi, N, requires_grad=False)
            polar = torch.linspace(0, 2 * math.pi, 2*N, requires_grad=False)
            azimuth, polar = torch.meshgrid(azimuth, polar)
            location = torch.cat([azimuth.reshape(-1, 1), polar.reshape(-1,
→ 1)], dim=1)
#             location = self.inner_sample()
            value = net(location) # predicted solution

            value = value.reshape((N, 2*N))

            fig, ax = plt.subplots(1, 1, figsize=(8, 6))
```

```

        sns.set()
        sns.heatmap(value.detach().numpy(), ax=ax, vmin=-0.5, vmax=0.5,
↪cbar=True)
        ax.set_xticks([])
        ax.set_yticks([])
        ax.set_title("Predicted Solution")
        plt.show()

    self.spherical_harmonic_transform(net)

    def spherical_harmonic_transform(self, net):
        azimuth = torch.linspace(0, math.pi, N, requires_grad=False)
        polar = torch.linspace(0, 2 * math.pi, 2*N, requires_grad=False)
        azimuth, polar = torch.meshgrid(azimuth, polar)
        location = torch.cat([azimuth.reshape(-1, 1), polar.reshape(-1,
↪1)], dim=1)
        function_grid = net(location)
        function_grid = function_grid.reshape((N, 2*N))

        coefficients = SHEExpandDH(function_grid.detach().numpy(),
↪sampling=2)
        nl = coefficients.shape[1]
        ls = np.arange(nl)[:10]

        power_per_l = spectrum(coefficients)[:10]
        fig, ax = plt.subplots(1, 1, figsize=(len(ls), 5))
        ax.plot(ls, power_per_l, 'bo')
        ax.plot(ls, power_per_l, 'k-')
        plt.xticks(range(len(ls)))
        ax.grid()
        plt.show()

    def plot_true(self):
        azimuth = torch.linspace(0, math.pi, N, requires_grad=False)
        polar = torch.linspace(0, 2 * math.pi, 2*N, requires_grad=False)
        azimuth, polar = torch.meshgrid(azimuth, polar)
        location = torch.cat([azimuth.reshape(-1, 1), polar.reshape(-1,
↪1)], dim=1)
        value = true_solution(location[:, 0:1], location[:, 1:])

        value = value.reshape((N, 2*N))

        fig, ax = plt.subplots(1, 1, figsize=(8, 6))
        sns.set()
        sns.heatmap(value.detach().numpy(), ax=ax, vmin=-0.5, vmax=0.5,
↪cbar=True)

```

```

        ax.set_xticks([])
        ax.set_yticks([])
        ax.set_title("True Solution")
        plt.show()

        self.sht_value(value)

    def sht_value(self, function_grid):
        function_grid = function_grid.reshape((N, 2*N))

        coefficients = SHExpandDH(function_grid.detach().numpy(),
→sampling=2)
        nl = coefficients.shape[1]
        ls = np.arange(nl)[:10]

        power_per_l = spectrum(coefficients)[:10]
        fig, ax = plt.subplots(1, 1, figsize=(len(ls), 5))
        ax.plot(ls, power_per_l, 'bo')
        ax.plot(ls, power_per_l, 'k-')
        plt.xticks(range(len(ls)))
        ax.grid()
        plt.show()

    def post_process(self, ax=None):
        if ax is None:
            for losses in self.loss_history:
                for i in range(3):
                    losses[i].detach().numpy()
            plt.plot(self.loss_history)
            plt.yscale('log')
            plt.legend(('BC loss', 'pde loss', 'predict error'))
            plt.show()
        else:
            for losses in self.loss_history:
                for i in range(3):
                    losses[i].detach().numpy()
            ax.plot(self.loss_history)
            ax.set_yscale('log')
            ax.set_ylim(1e-4, 100)
            ax.legend(('BC loss', 'pde loss', 'predict error'))

    def predict_error(self):
        coor = self.inner_sample()
        true = self.problem.ground_truth(coor)
        predict = self.net(coor)
        predict_error = self.pde_loss_f(true, predict)
        return predict_error

```



```

def train(self, ax100=None):
    self.plot_true()
    problem = self.problem
    net = self.net
    opt = self.opt
    maxiter = self.maxiter

    for iter in range(maxiter):
        net.zero_grad()

        coor_inner = self.inner_sample().detach().requires_grad_(True)
        # coor_inner = self.inner_sample().requires_grad_(True)
        infer_value_inner = net(coor_inner)
        # truth_inner, predict_inner = problem.pde(coor_inner,
        # infer_value_inner)
        predict_inner, truth_inner = problem.pde(coor_inner,
        # infer_value_inner)
        ## order, is the lhs even the predicted value???
        ## reason the loss function is going up could be because we are
        # computing the negative gradient, which could happen by switching the order
        # of y, y_pred in orward and backward functions of the loss.

        self.pde_loss = self.pde_loss_f(predict_inner, truth_inner)

        bc_samples = self.bc_sample()
        if bc_samples is None:
            self.bc_loss = torch.tensor(0.)
        else:
            coor_bc = bc_samples.detach().requires_grad_(True)
            infer_value_bc = net(coor_bc)
            truth_bc, predict_bc = problem.bound_condition(coor_bc,
            # infer_value_bc)
            self.bc_loss = self.bc_loss_f(predict_bc, truth_bc)

            init_samples = self.init_sample()
            if init_samples is None:
                self.init_loss = torch.tensor(0.)
            else:
                coor_init = init_samples.detach().requires_grad_(True)
                infer_value_init = net(coor_init)
                truth_init, predict_init = problem.
            # bound_condition(coor_init, infer_value_init)
                self.init_loss = self.bc_loss_f(predict_init, truth_init)

        self.predict_error_value = self.predict_error()
        self.total_loss = self.pde_loss + self.bc_loss + self.init_loss

```

```

        self.add_loss_history()
        self.total_loss.register_hook(lambda grad: print('gradient', ↵
↵grad))

        self.total_loss.backward()

        opt.step()
        opt.zero_grad()
        # scheduler.step()

        if iter % (maxiter // 20) == 0:
            print("iteration {}: loss = {}, pde loss = {}".format(iter, ↵
↵self.total_loss, self.pde_loss))

            if iter % int(maxiter / 5) == 0:
                self.plot(net)

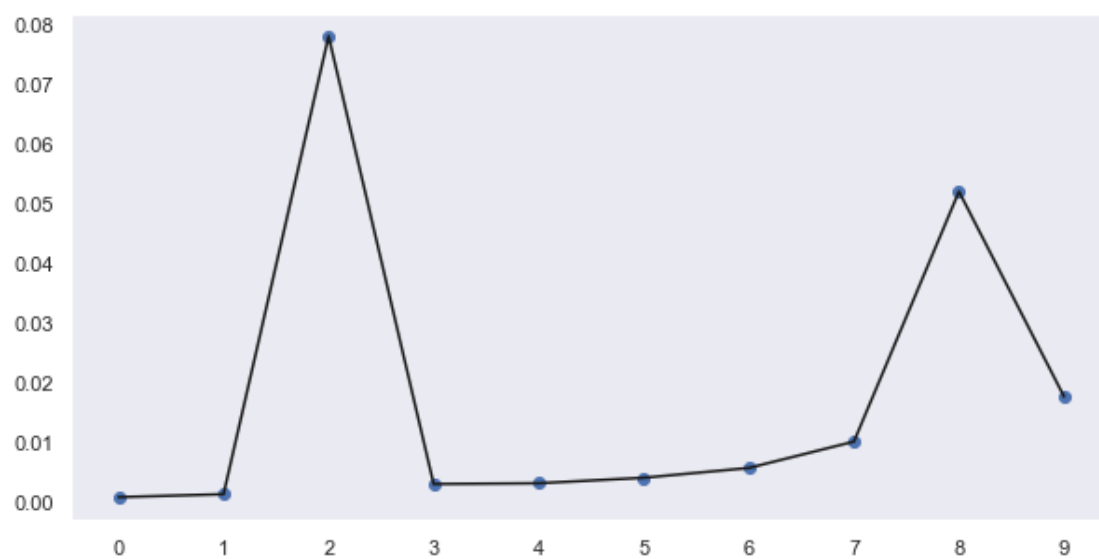
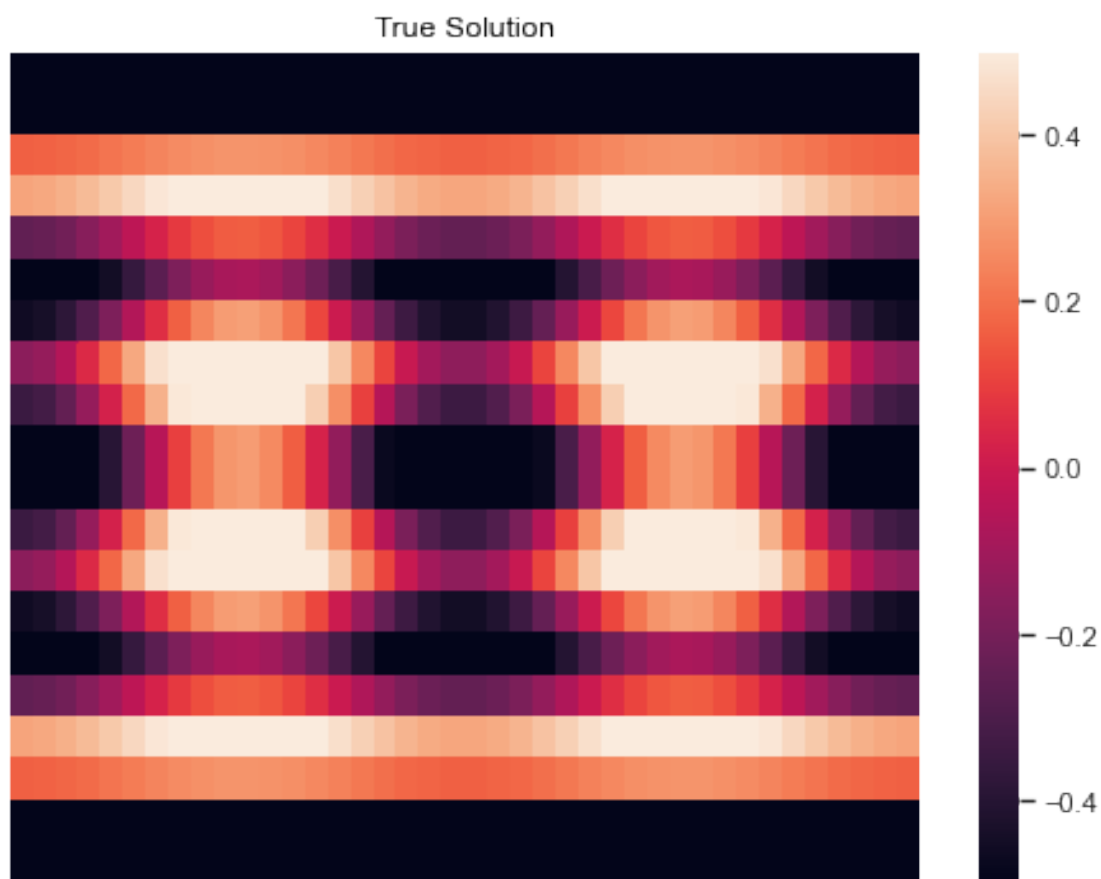
        self.plot(net)

        # self.post_process()

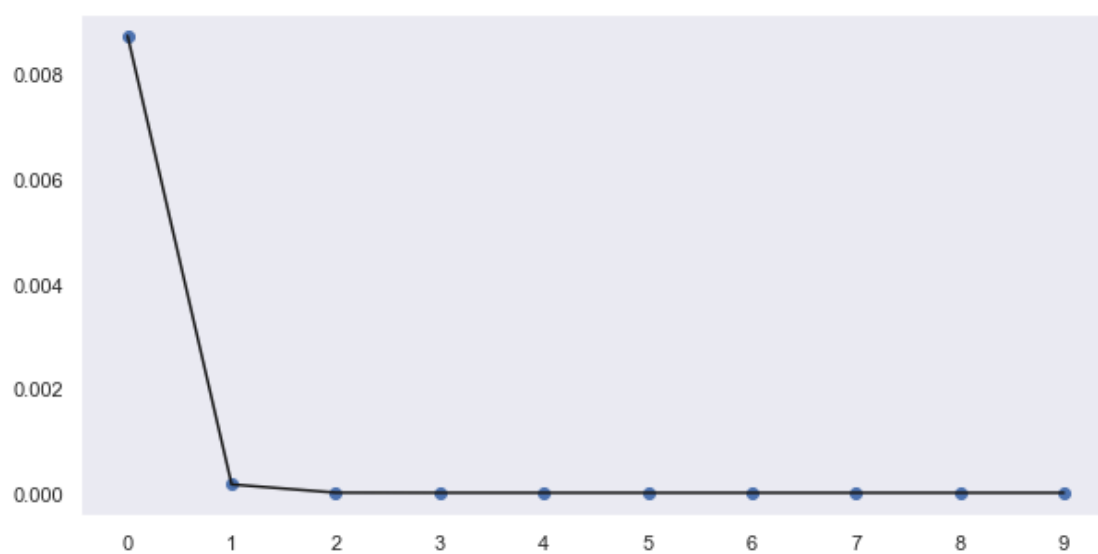
    return Poisson1dModel()

model = construct_model(Sphere_Net([3, 150, 50, 50, 50, 16, 1]))
model.train()

```

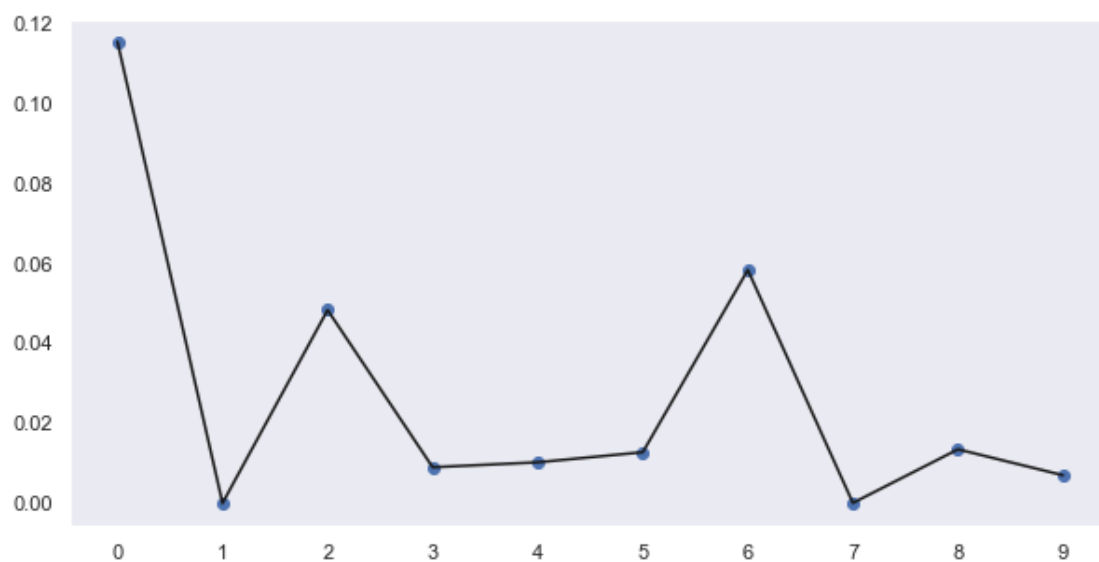
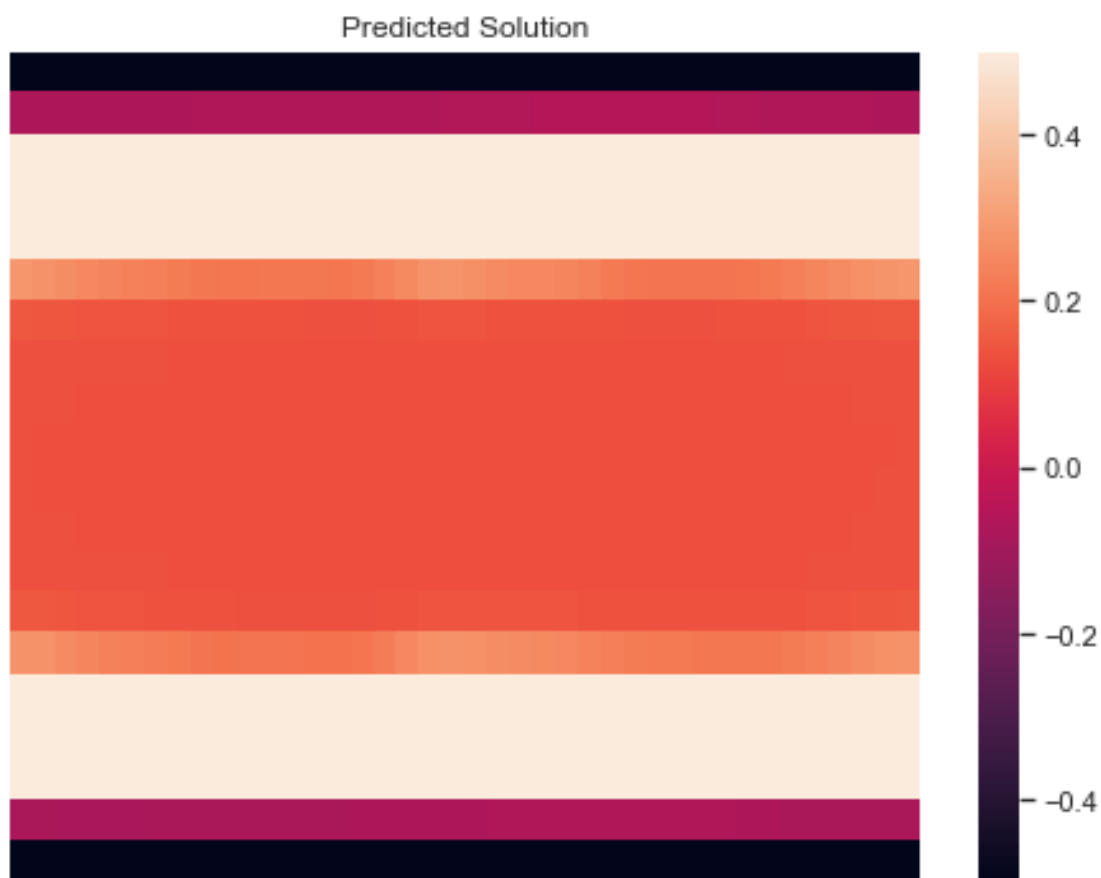


iteration 0: loss = 1223.7763671875, pde loss = 1223.727783203125

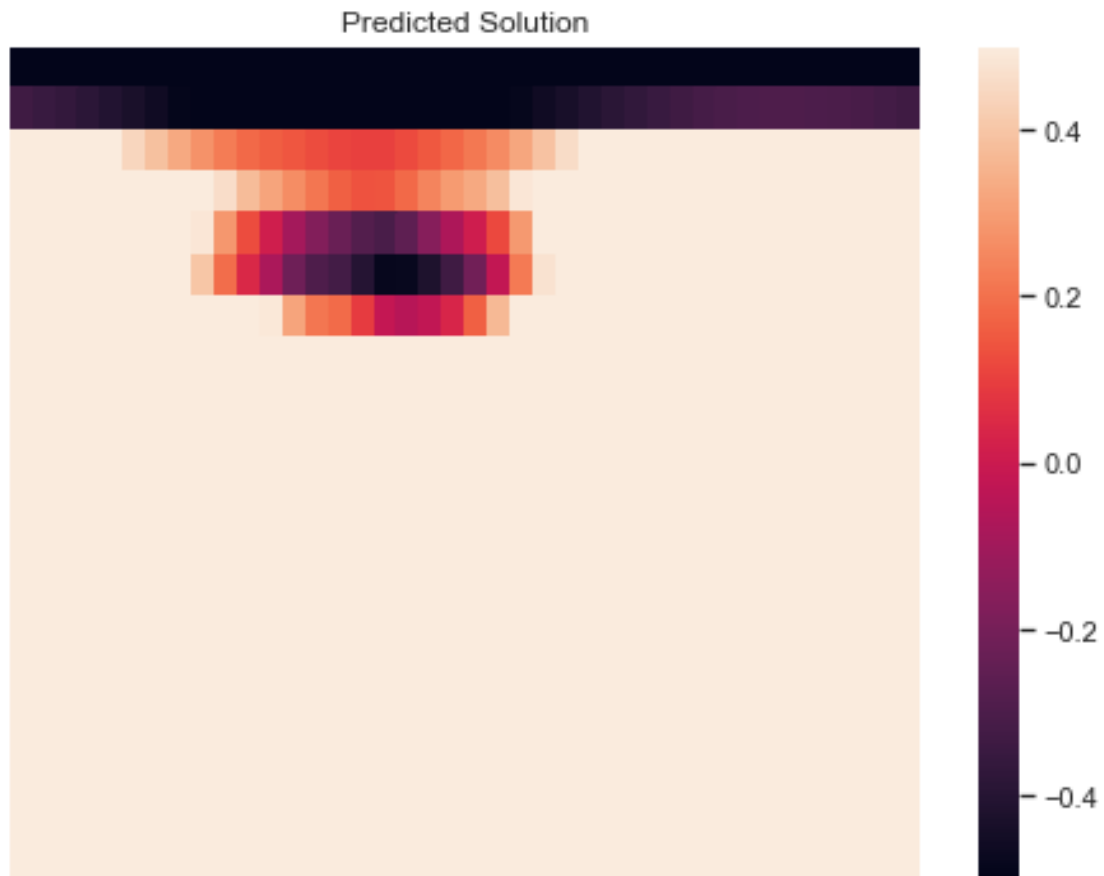


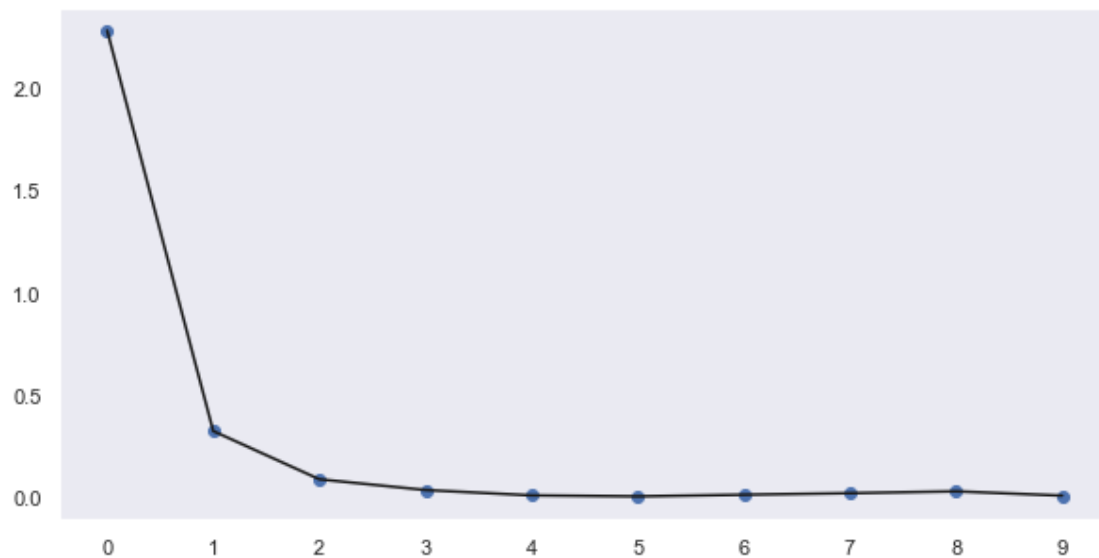
iteration 500: loss = 173.2105712890625, pde loss = 171.8254852294922

iteration 1000: loss = 98.76496124267578, pde loss = 98.71331787109375  
iteration 1500: loss = 98.11726379394531, pde loss = 98.1082992553711  
iteration 2000: loss = 97.98534393310547, pde loss = 97.98443603515625

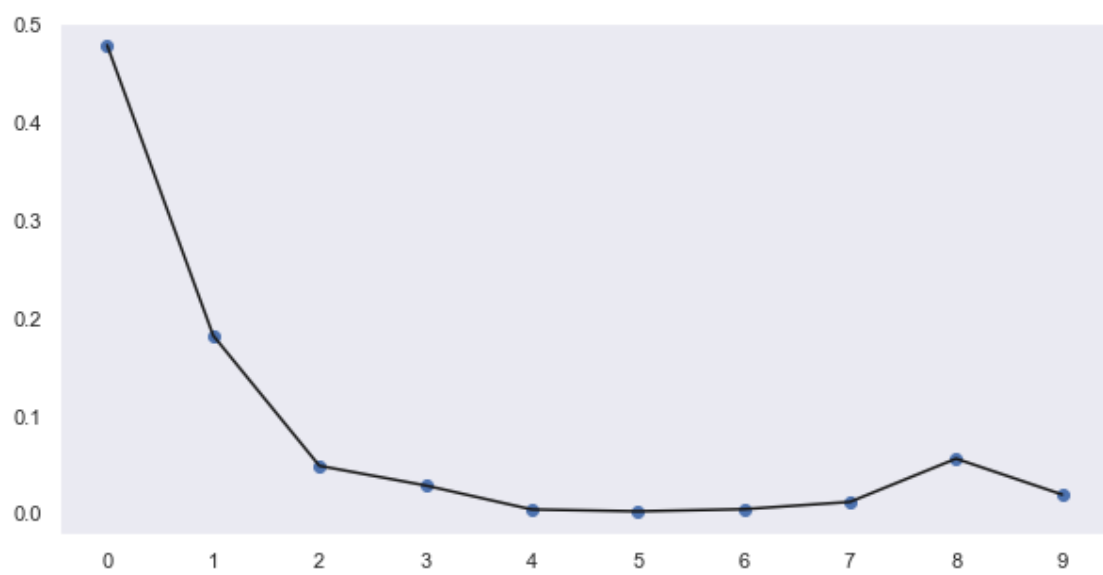
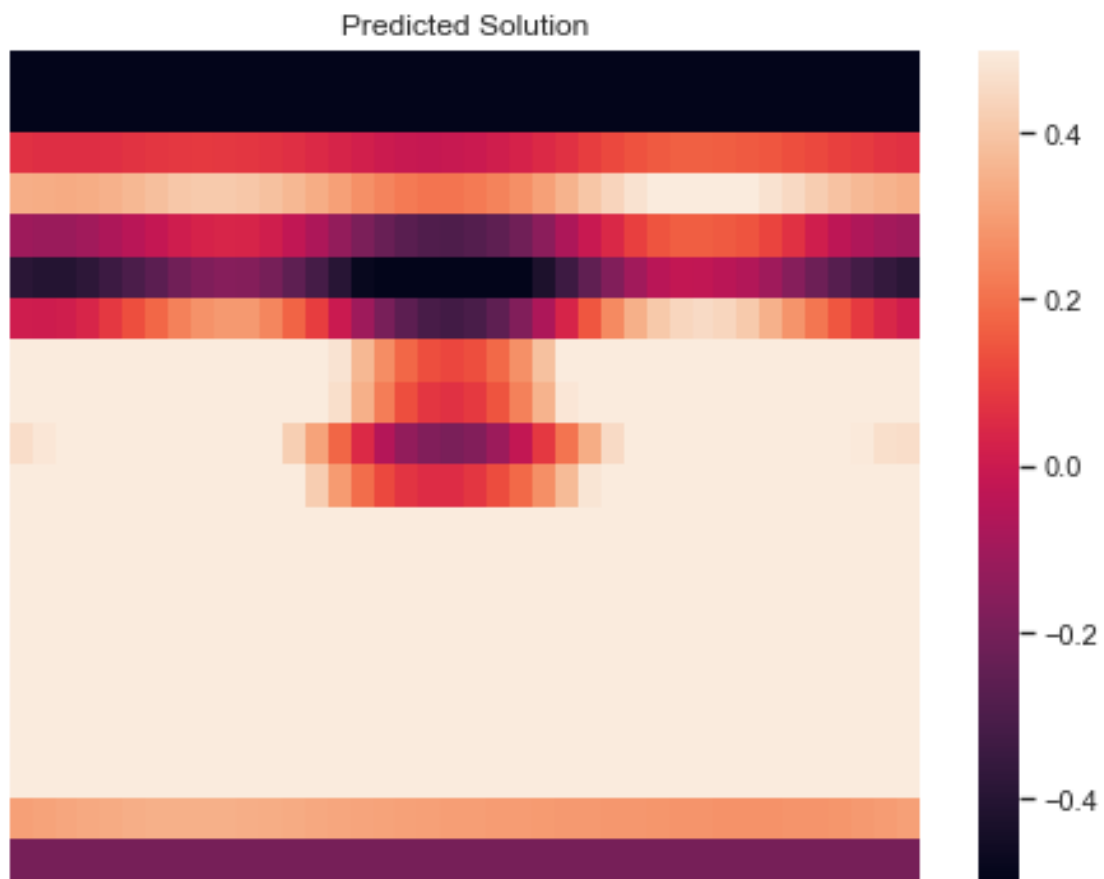


```
iteration 2500: loss = 97.85435485839844, pde loss = 97.85429382324219
iteration 3000: loss = 97.6225814819336, pde loss = 97.6225814819336
iteration 3500: loss = 290.9342956542969, pde loss = 288.67242431640625
iteration 4000: loss = 76.10307312011719, pde loss = 75.6131591796875
```



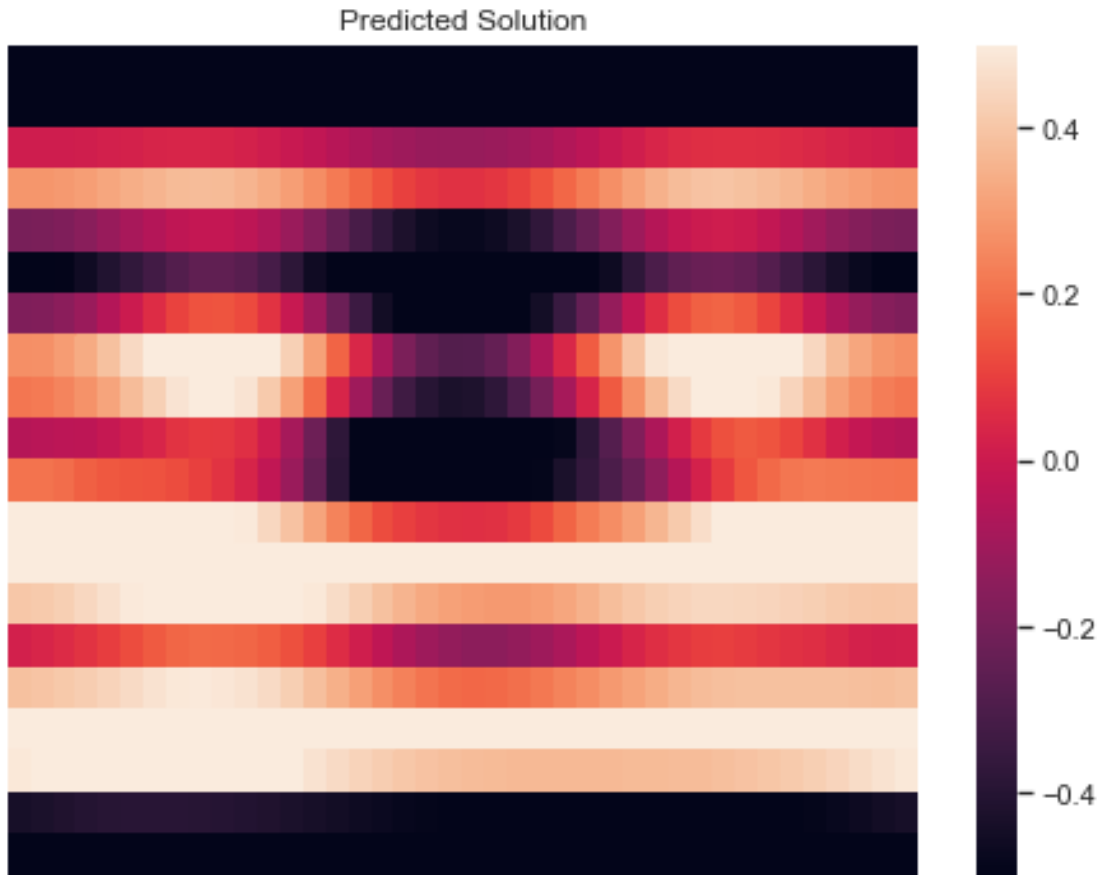


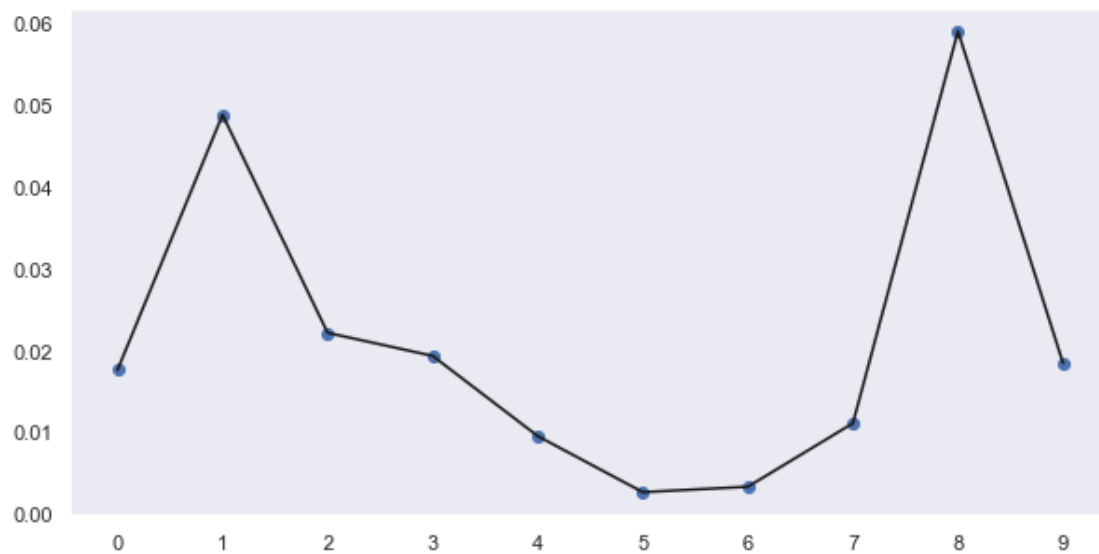
```
iteration 4500: loss = 7.760080814361572, pde loss = 7.669693470001221
iteration 5000: loss = 3.8647916316986084, pde loss = 3.771446466445923
iteration 5500: loss = 2.1373307704925537, pde loss = 2.1143269538879395
iteration 6000: loss = 1.1158379316329956, pde loss = 1.0956239700317383
```



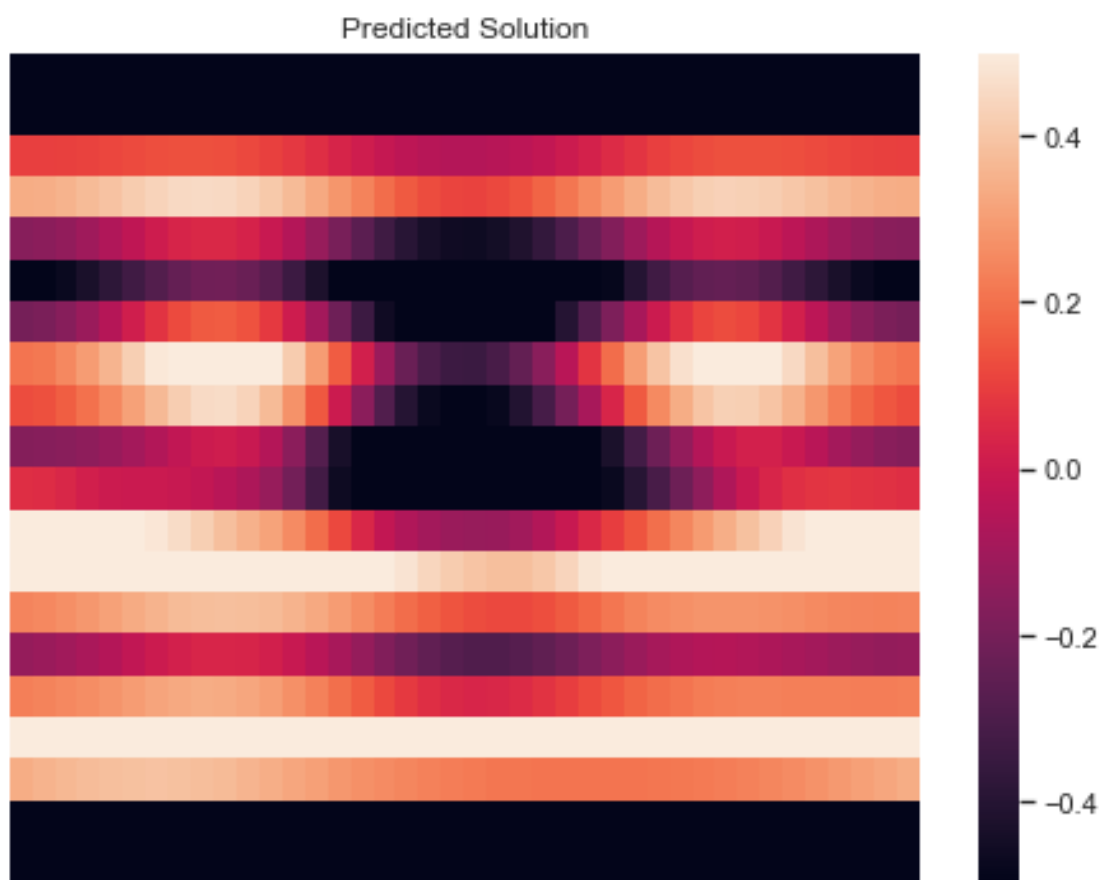


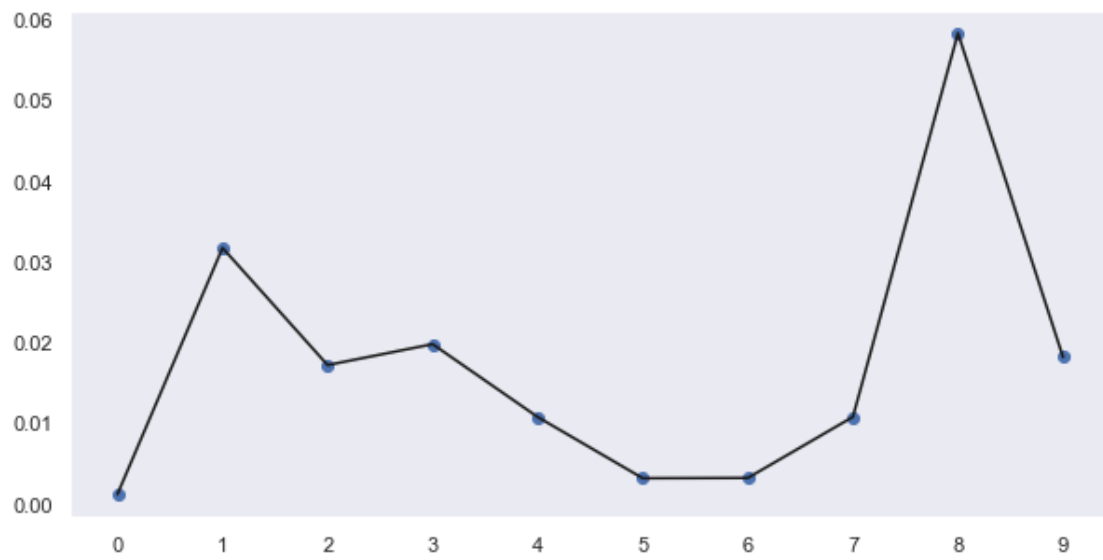
iteration 6500: loss = 0.7061406373977661, pde loss = 0.6938017010688782  
iteration 7000: loss = 0.5710601210594177, pde loss = 0.568142831325531  
iteration 7500: loss = 0.5270670652389526, pde loss = 0.5224175453186035  
iteration 8000: loss = 0.17385658621788025, pde loss = 0.17270122468471527



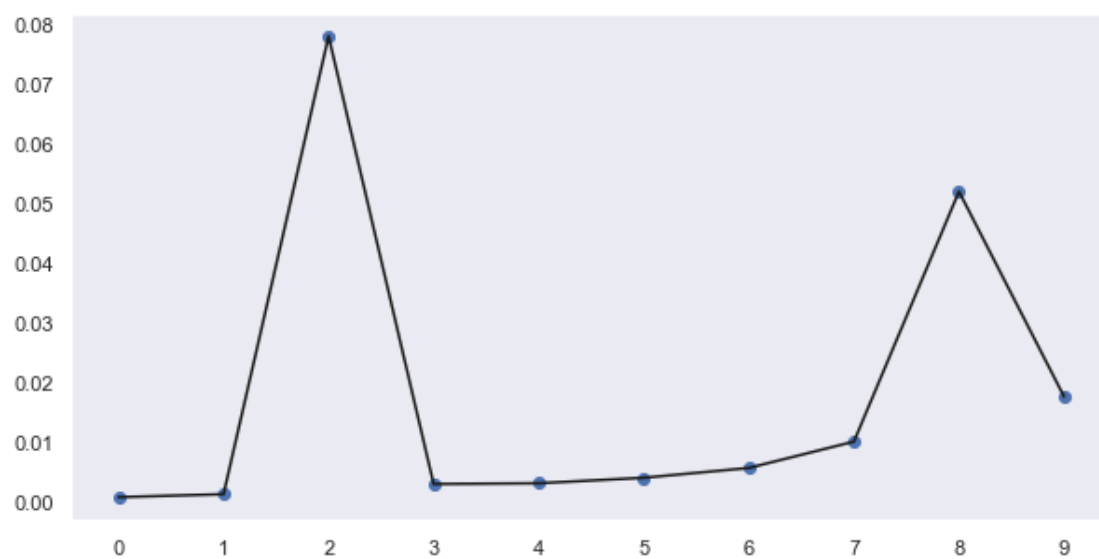
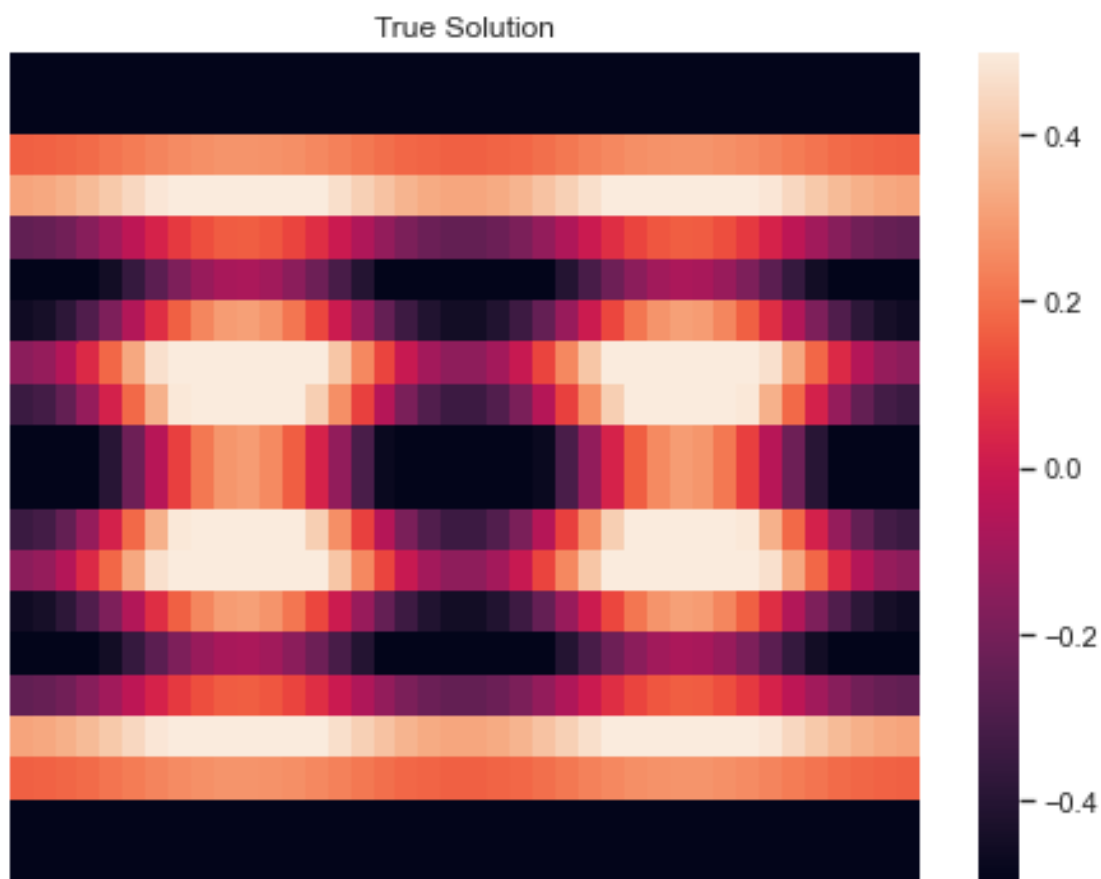


iteration 8500: loss = 0.34547728300094604, pde loss = 0.3417627215385437  
iteration 9000: loss = 0.1090117022395134, pde loss = 0.10857059806585312  
iteration 9500: loss = 0.15040071308612823, pde loss = 0.14941415190696716

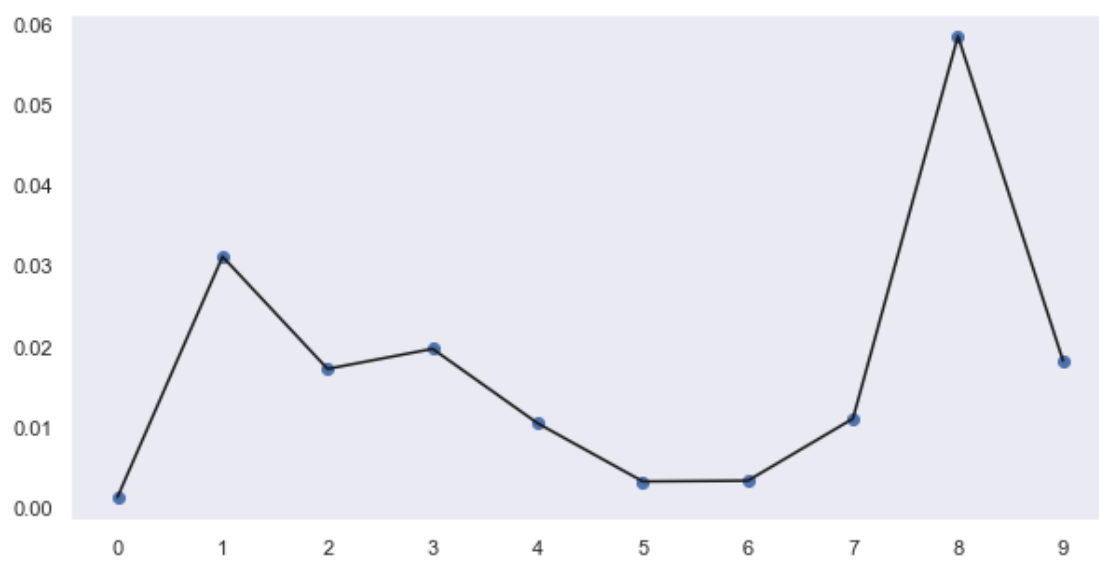
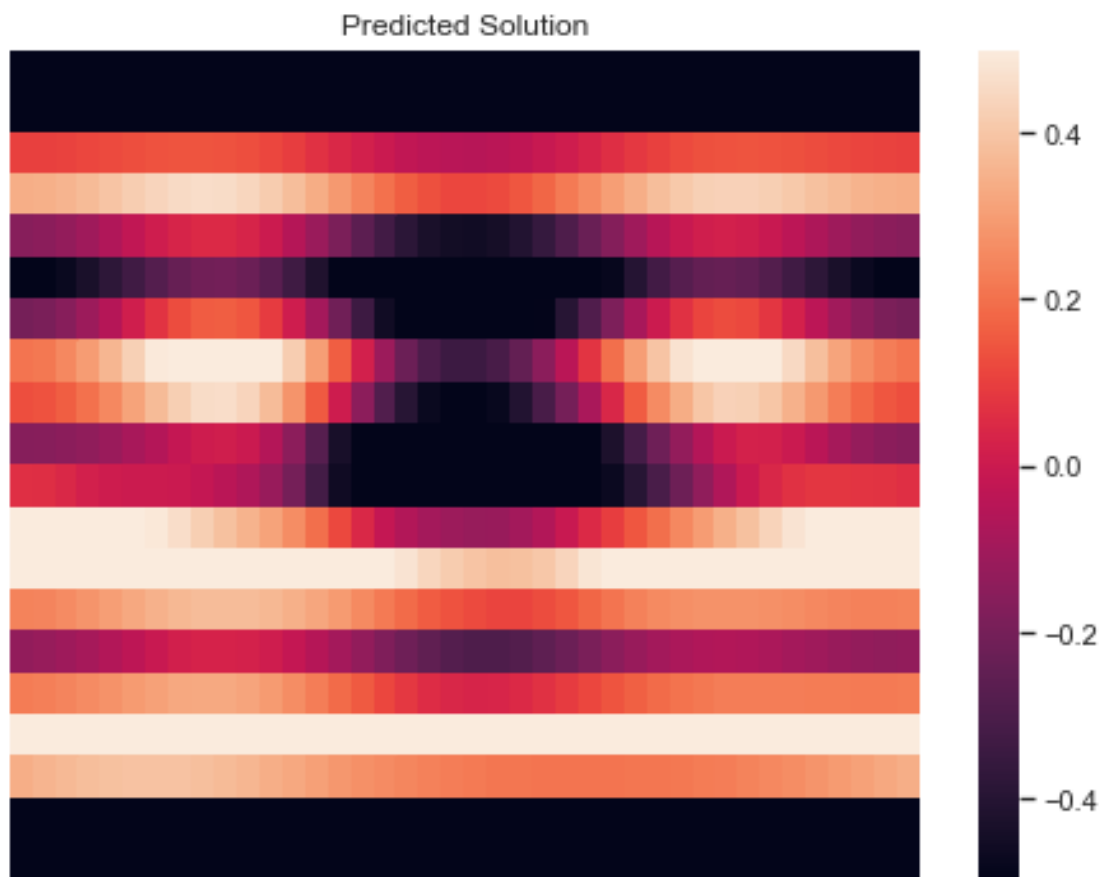




```
[5]: model.train()
```

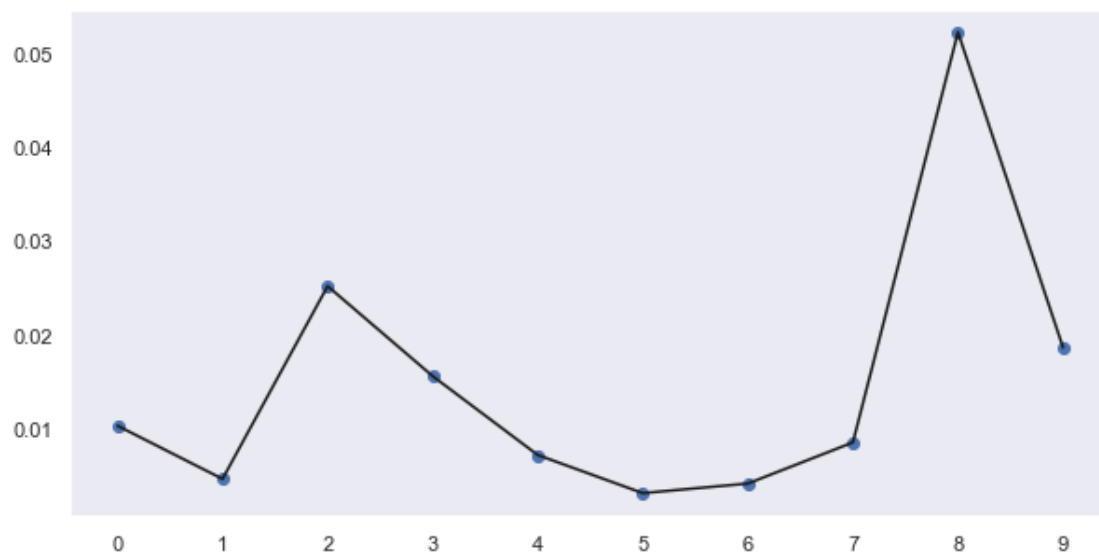
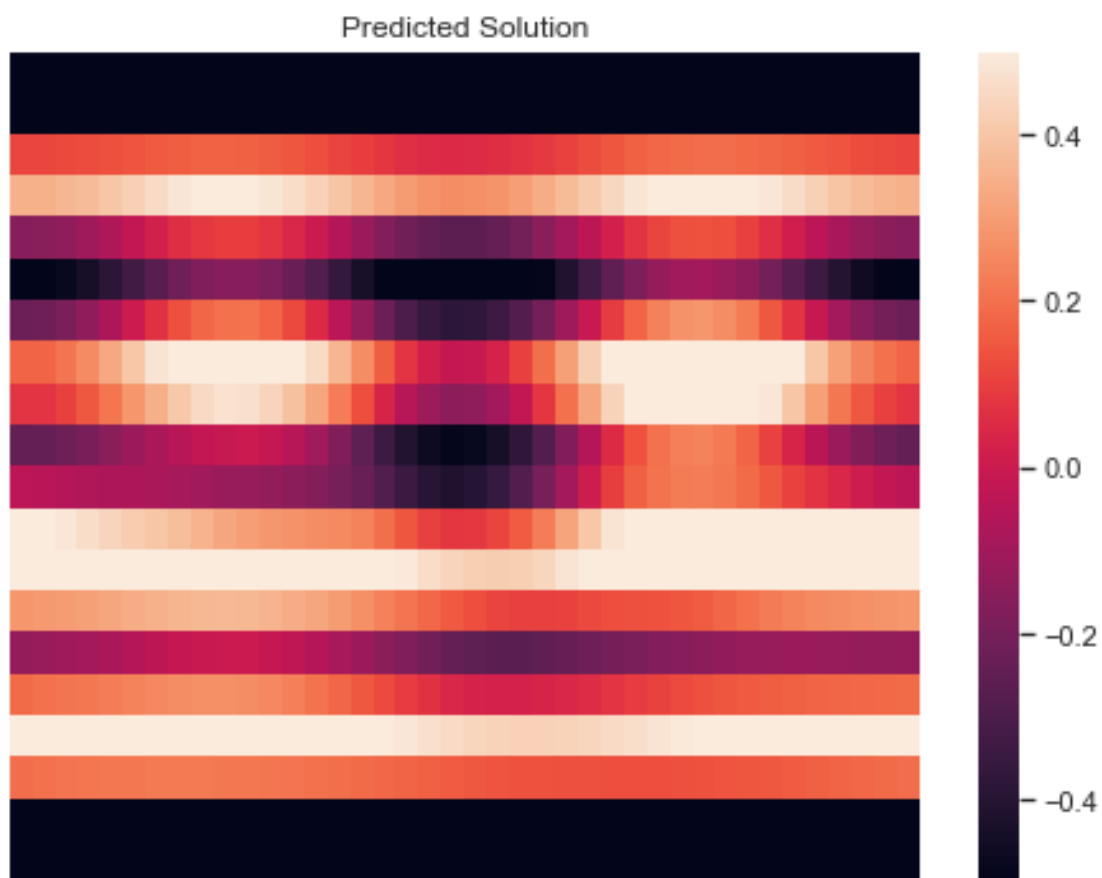


iteration 0: loss = 0.049360763281583786, pde loss = 0.04936075210571289

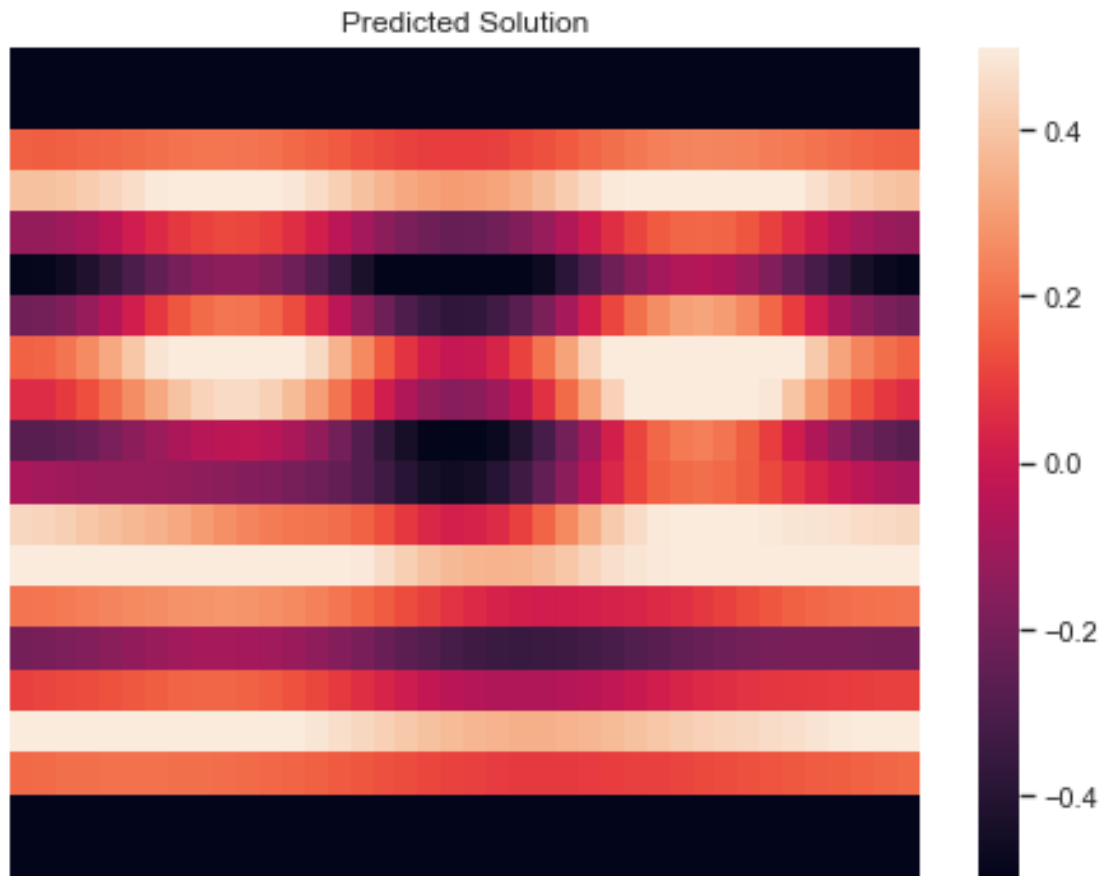


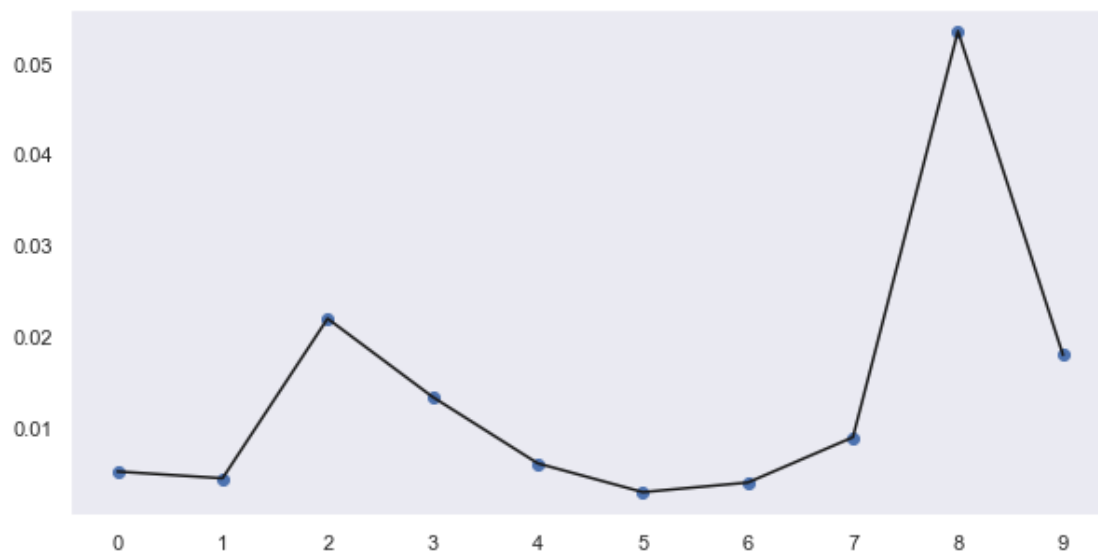
iteration 500: loss = 0.2926311492919922, pde loss = 0.2905862331390381

iteration 1000: loss = 46.19804382324219, pde loss = 45.93537902832031  
iteration 1500: loss = 0.2615647315979004, pde loss = 0.2615553140640259  
iteration 2000: loss = 0.09539905190467834, pde loss = 0.09539737552404404



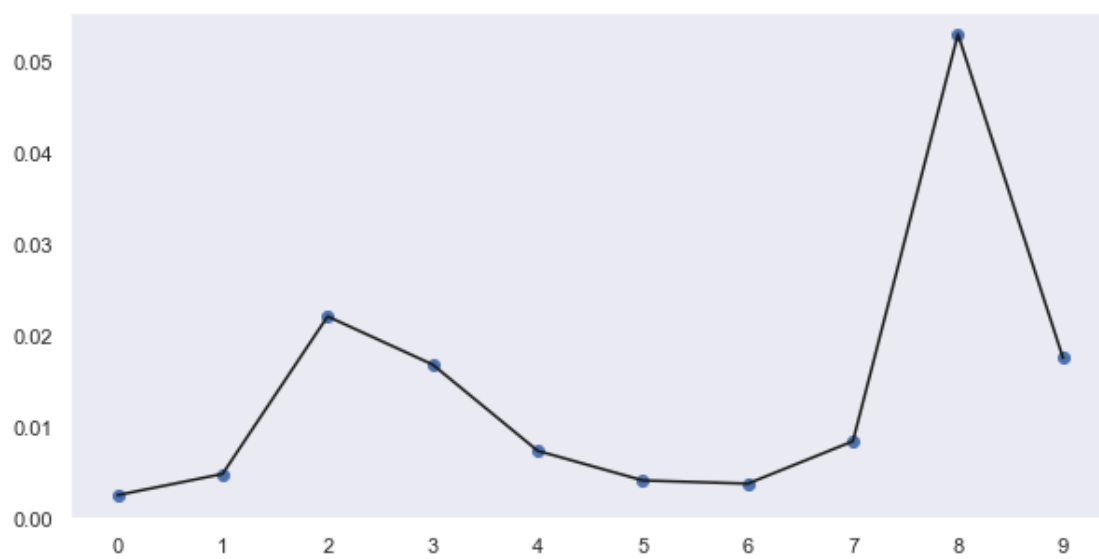
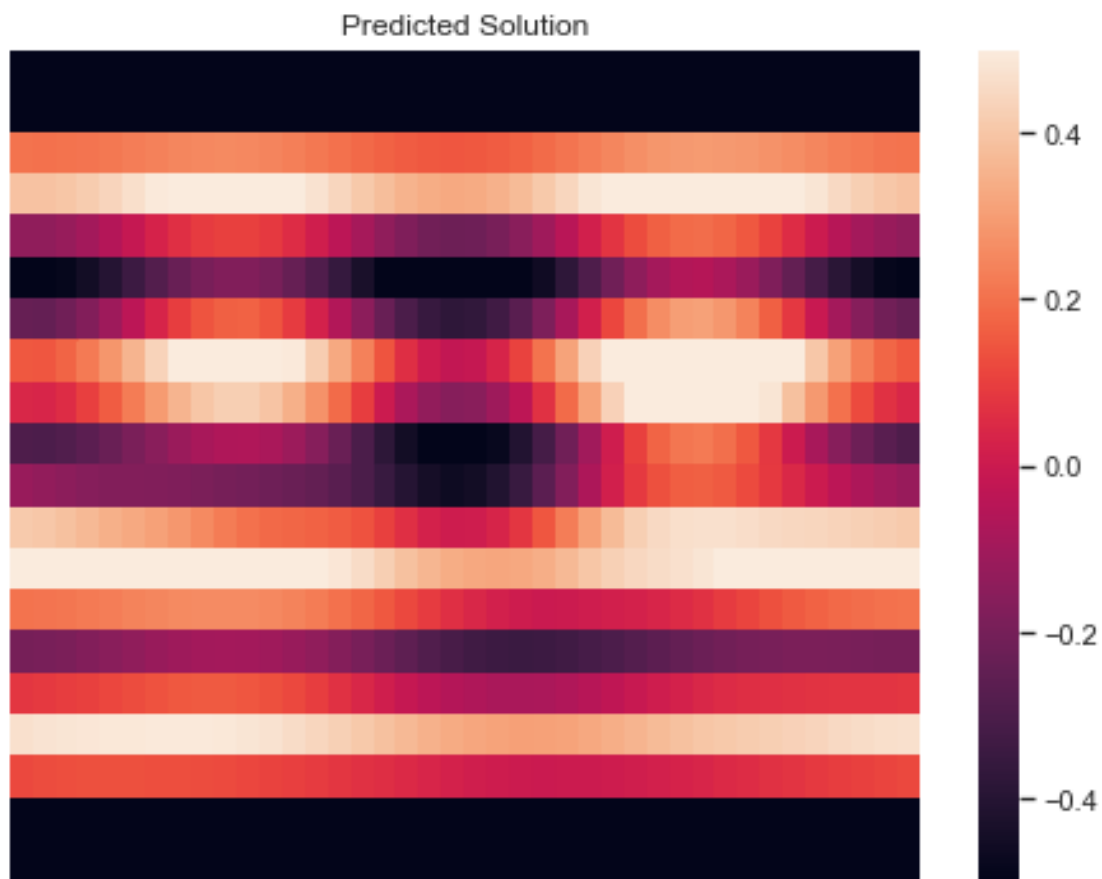
```
iteration 2500: loss = 0.09953085333108902, pde loss = 0.09952633082866669
iteration 3000: loss = 0.03756396844983101, pde loss = 0.03756355866789818
iteration 3500: loss = 0.19160115718841553, pde loss = 0.19150884449481964
iteration 4000: loss = 0.10312749445438385, pde loss = 0.10300275683403015
```





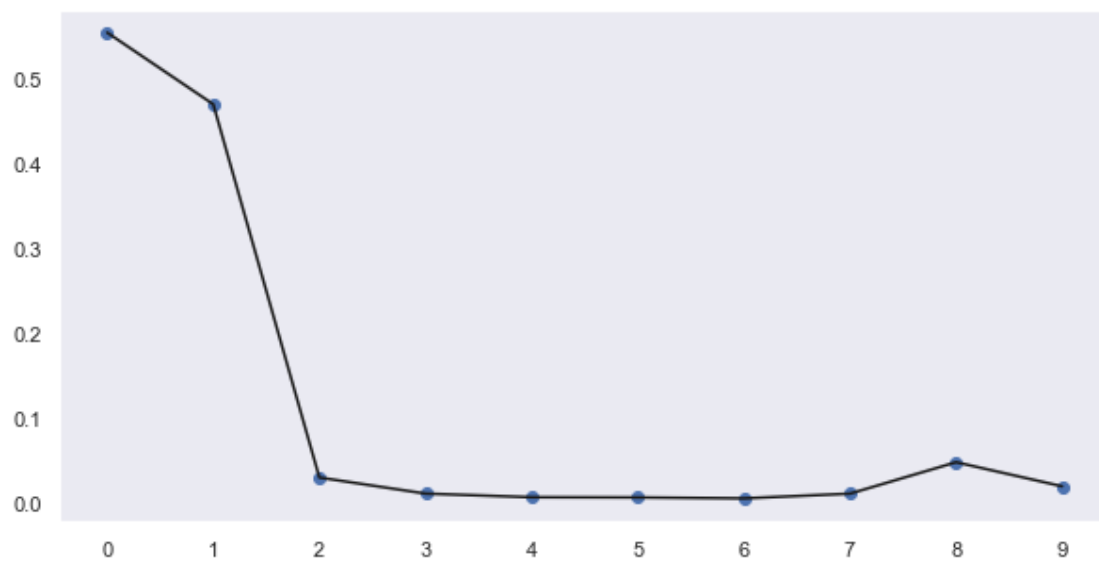
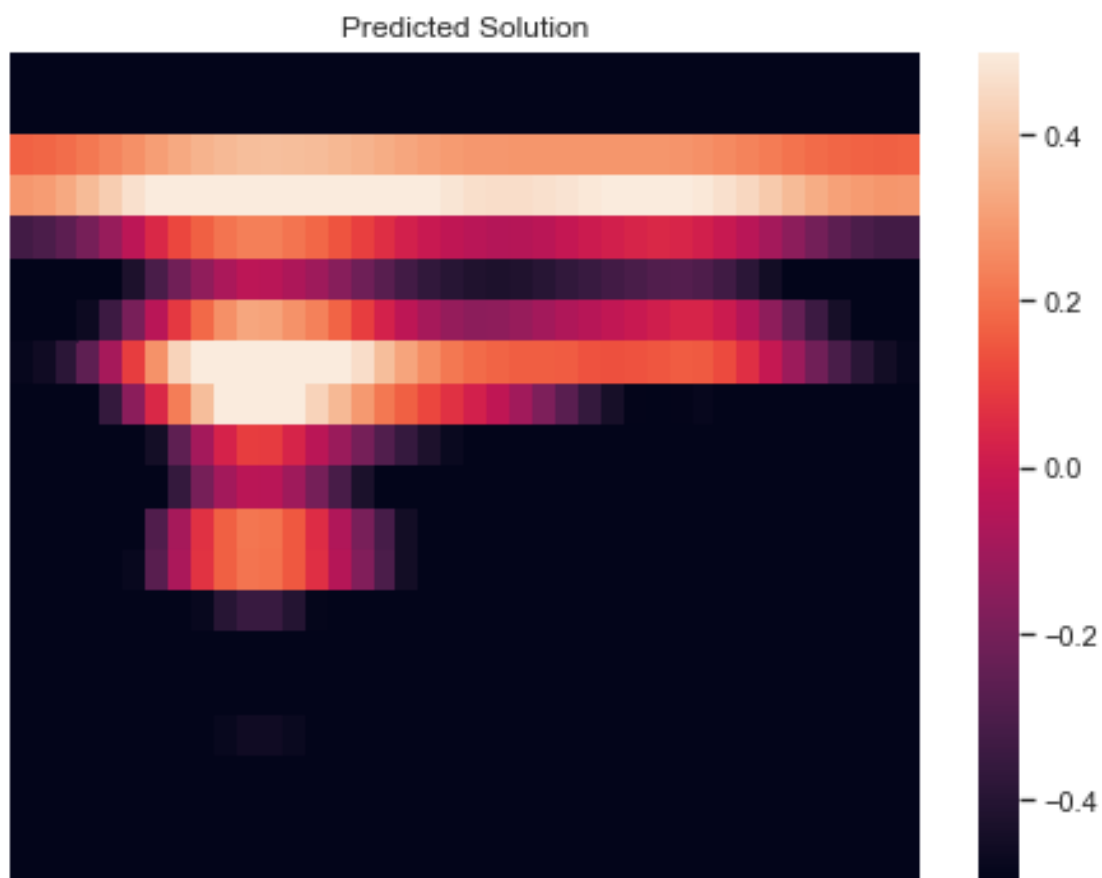
```
iteration 4500: loss = 0.03741665929555893, pde loss = 0.03741537034511566
iteration 5000: loss = 0.05580885335803032, pde loss = 0.05508812516927719
iteration 5500: loss = 0.018253987655043602, pde loss = 0.018230661749839783
iteration 6000: loss = 0.12178760766983032, pde loss = 0.12112918496131897
```



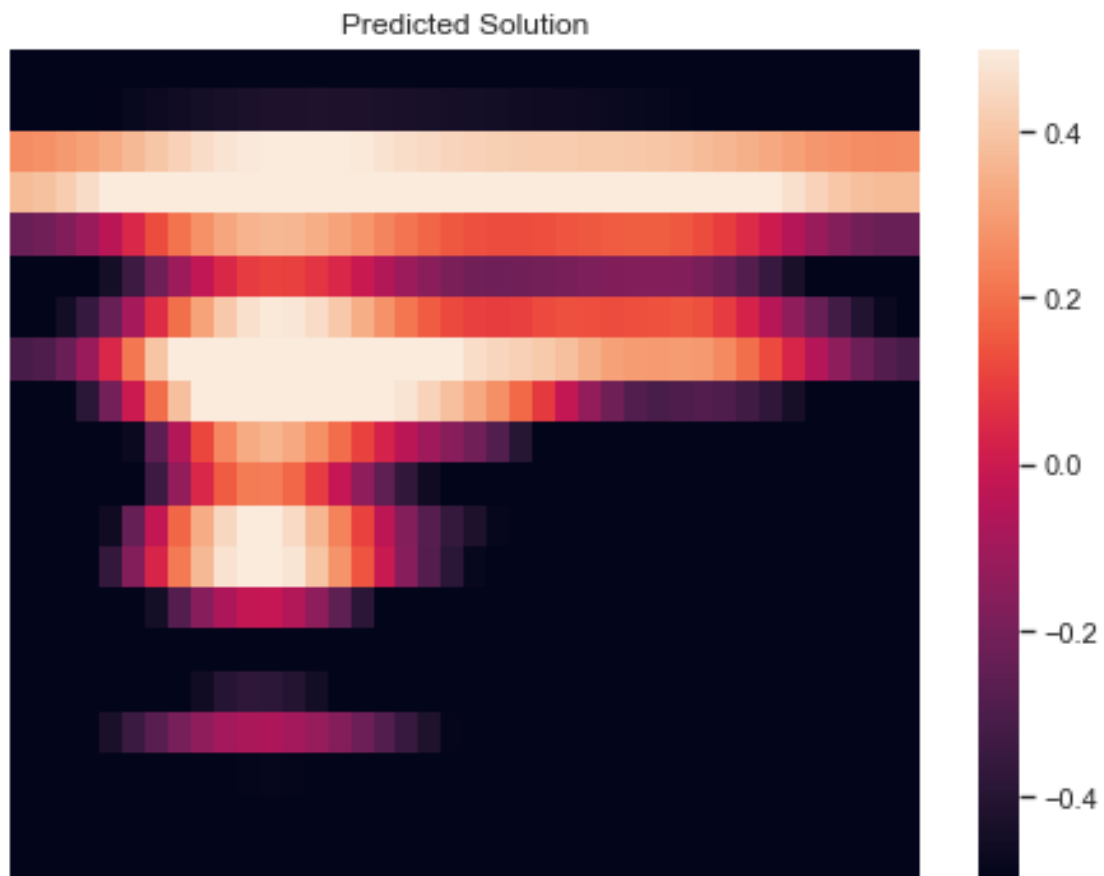


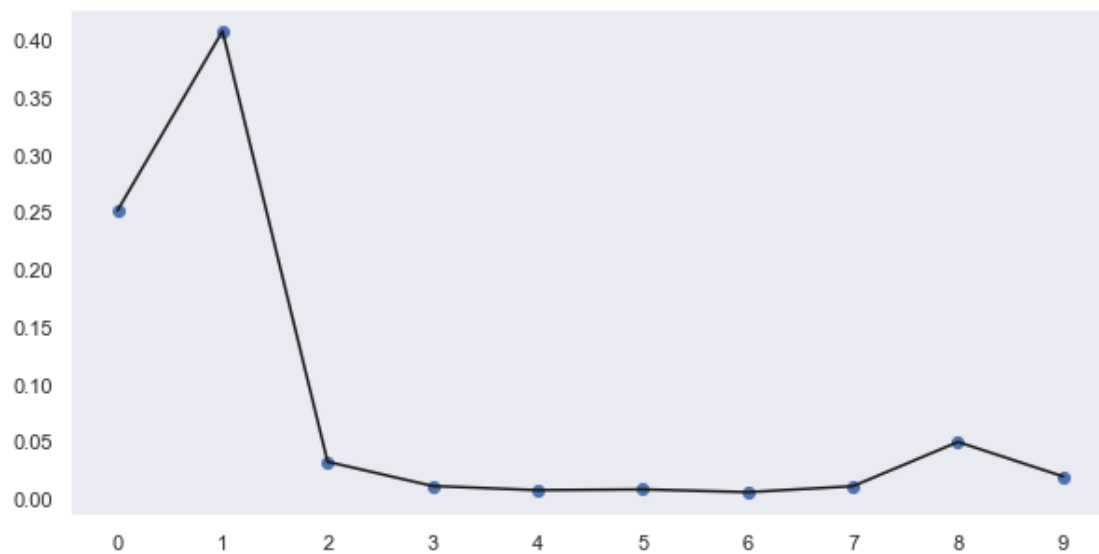
iteration 6500: loss = 0.021924365311861038, pde loss = 0.021814607083797455

iteration 7000: loss = 0.01268573198467493, pde loss = 0.012683620676398277  
iteration 7500: loss = 2.0480430126190186, pde loss = 1.946489691734314  
iteration 8000: loss = 0.4135385751724243, pde loss = 0.4033425450325012

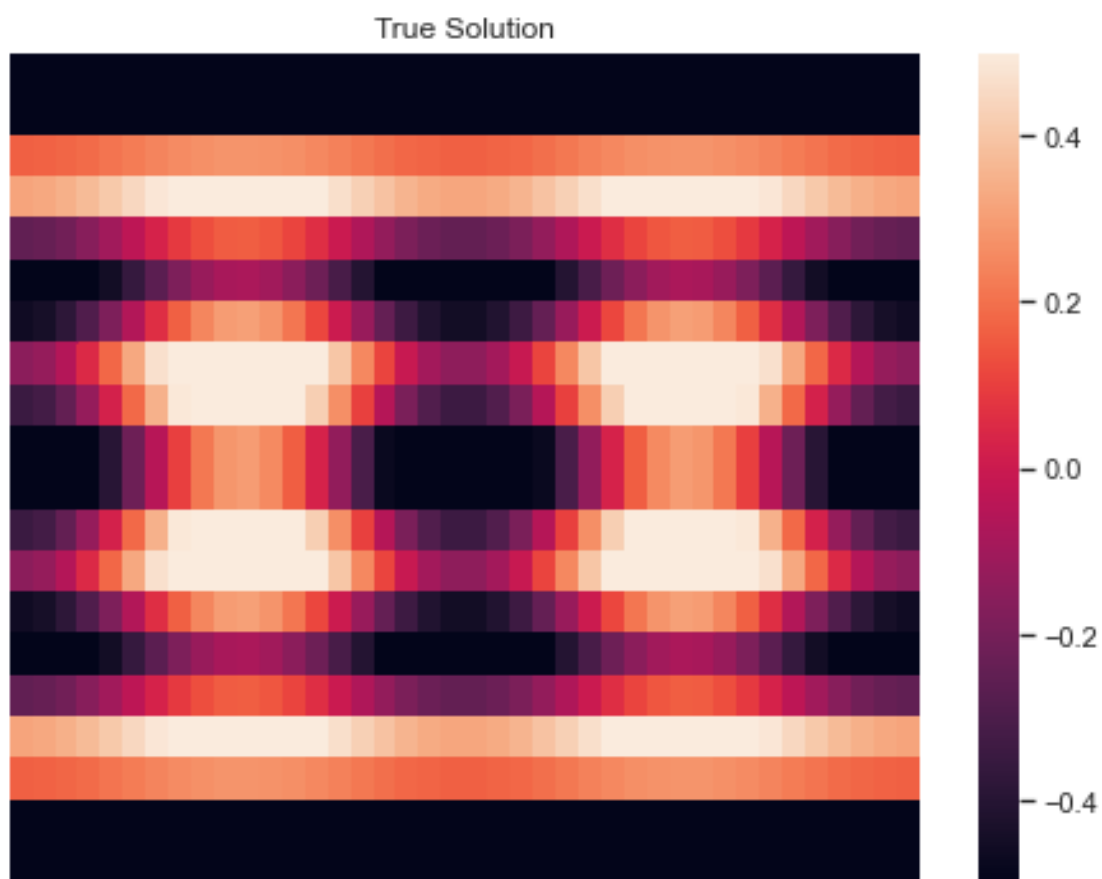


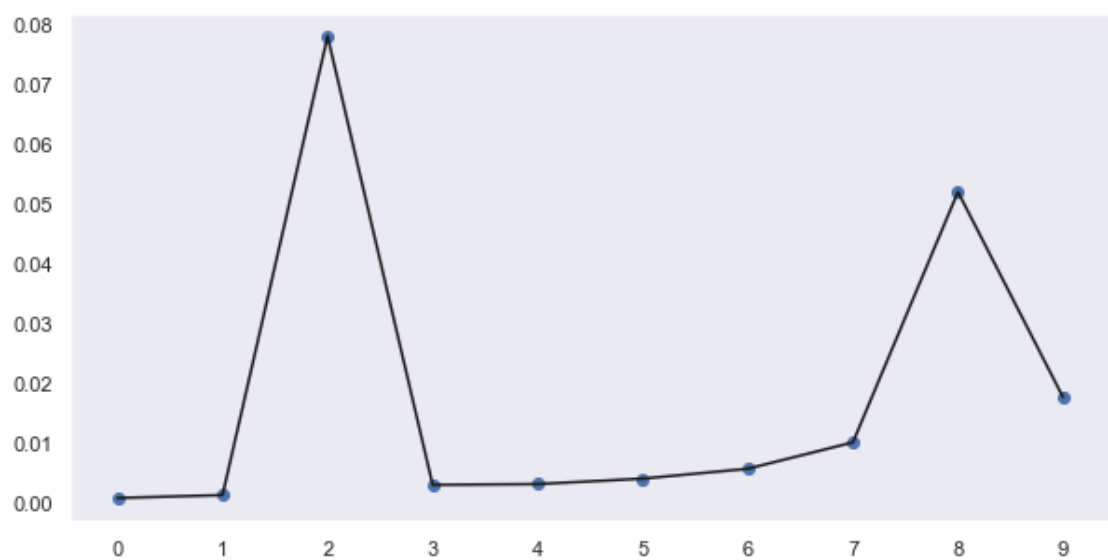
iteration 8500: loss = 0.16687560081481934, pde loss = 0.166449368000003052  
iteration 9000: loss = 0.19622446596622467, pde loss = 0.1961785852909088  
iteration 9500: loss = 0.08773330599069595, pde loss = 0.08770827203989029



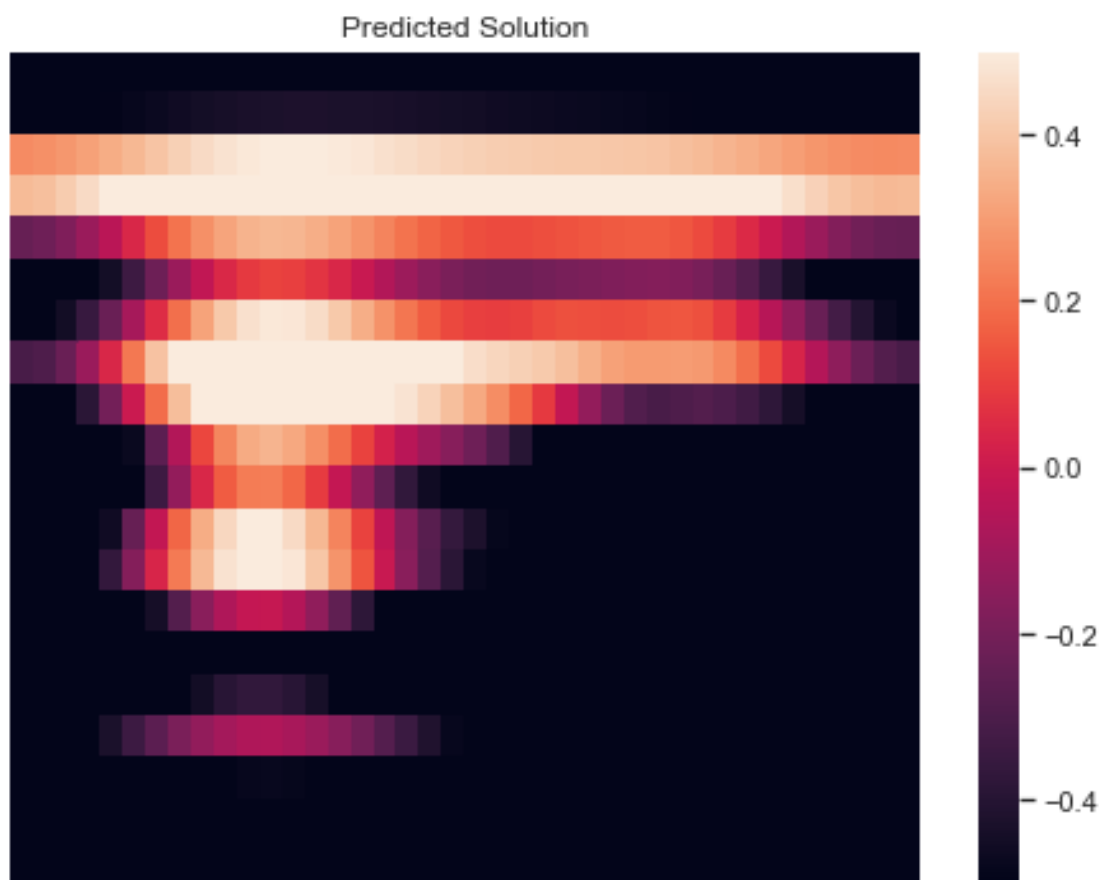


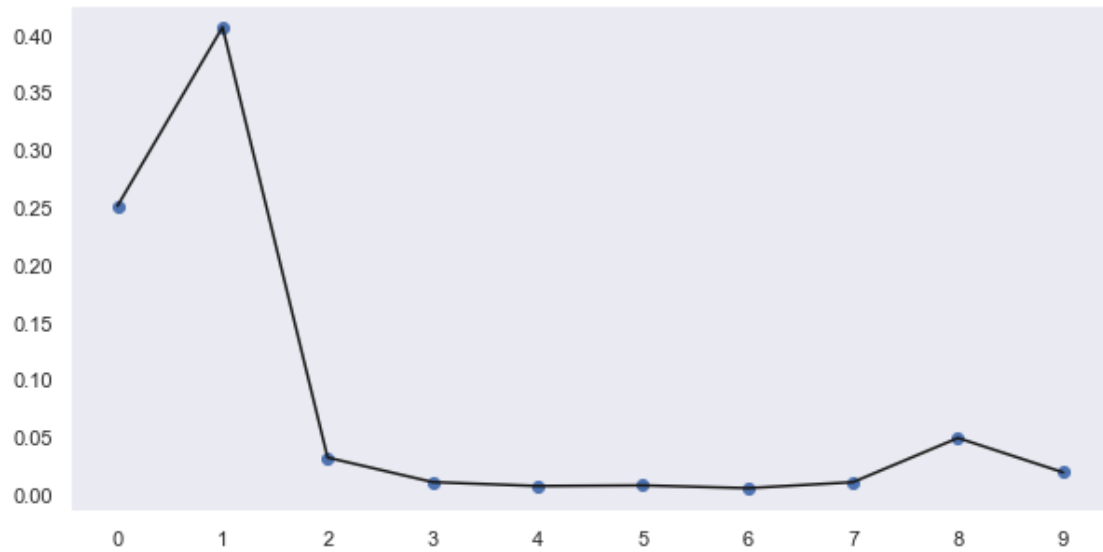
```
[6]: model.train()
```



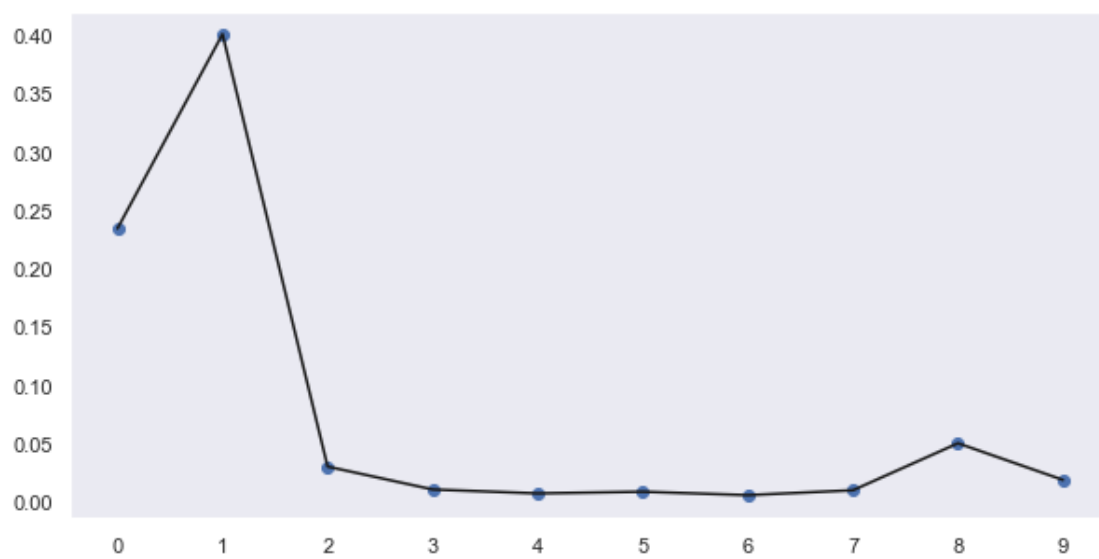
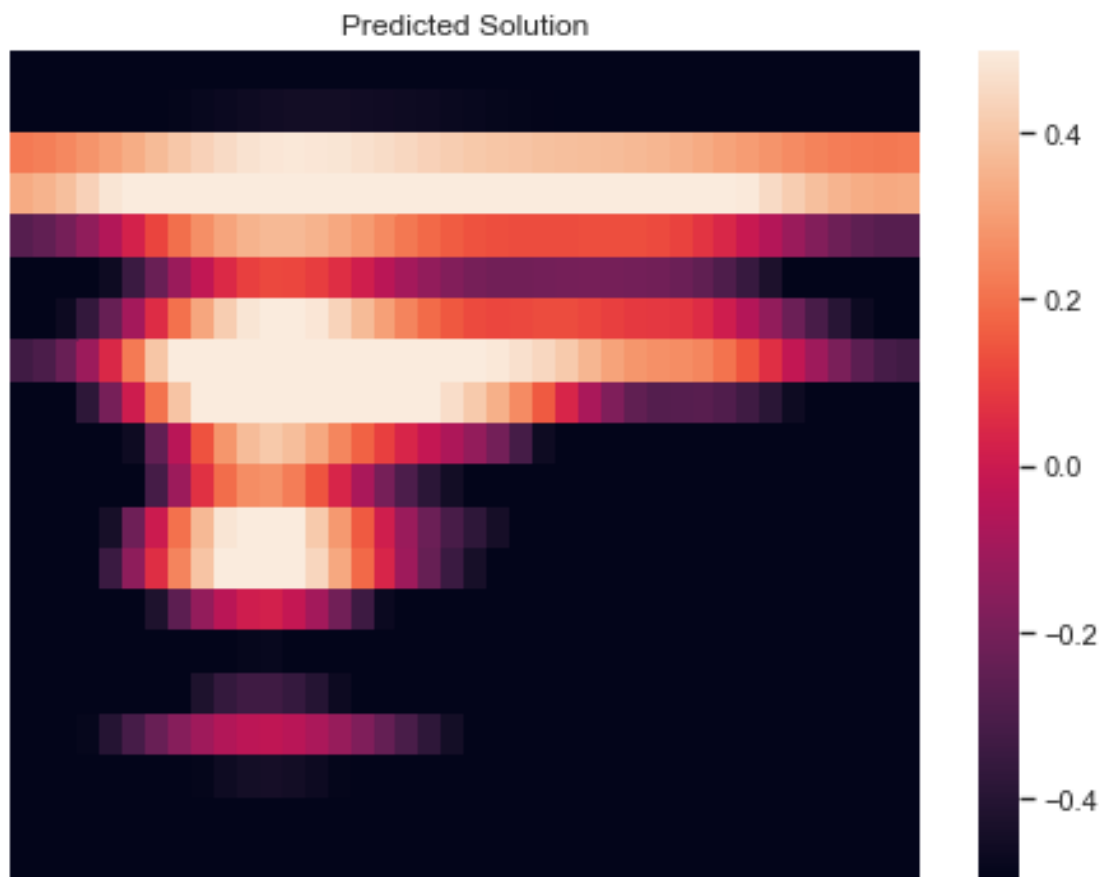


iteration 0: loss = 0.05453793331980705, pde loss = 0.054532334208488464



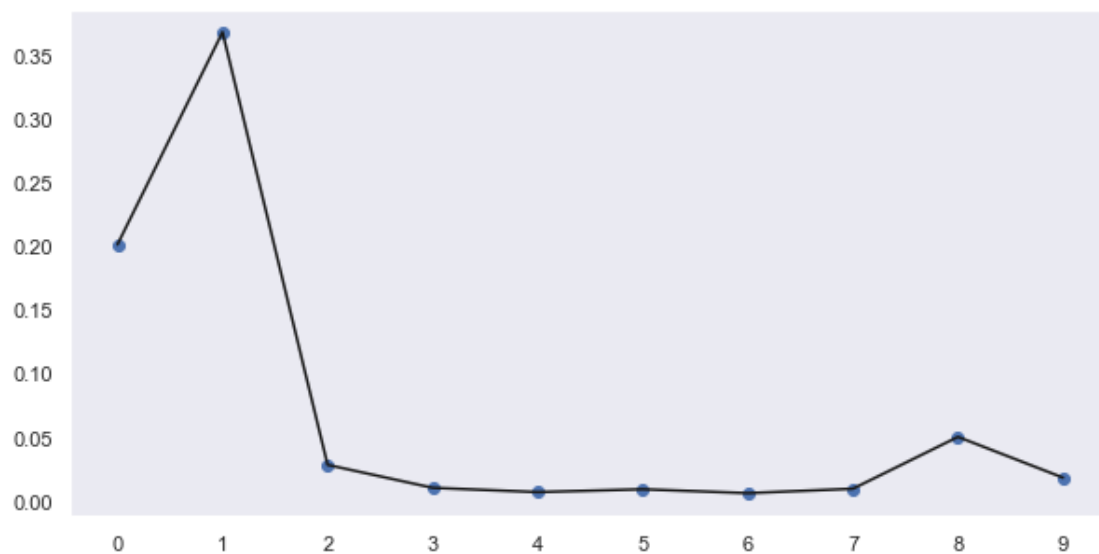
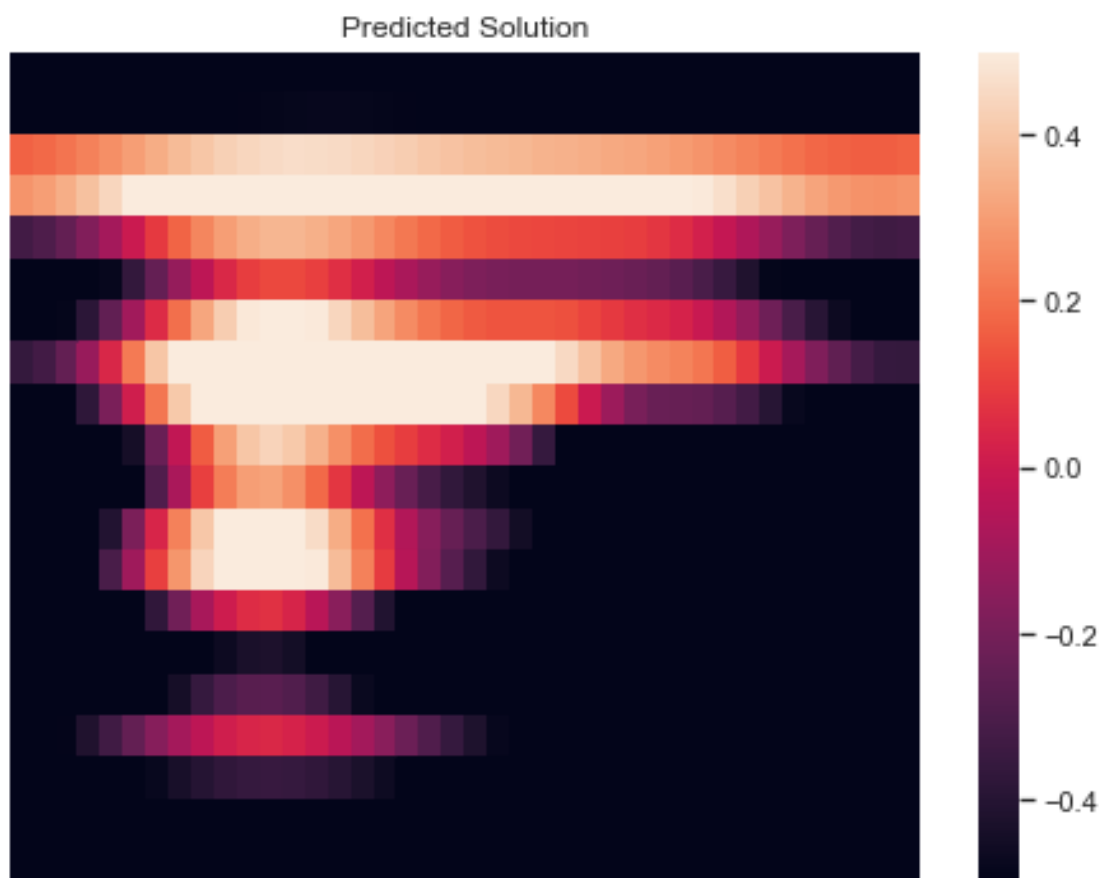


```
iteration 500: loss = 0.10810568928718567, pde loss = 0.10807281732559204
iteration 1000: loss = 0.074444817572832108, pde loss = 0.07444412261247635
iteration 1500: loss = 0.0648028627038002, pde loss = 0.06478368490934372
iteration 2000: loss = 0.0524415448307991, pde loss = 0.05244114249944687
```



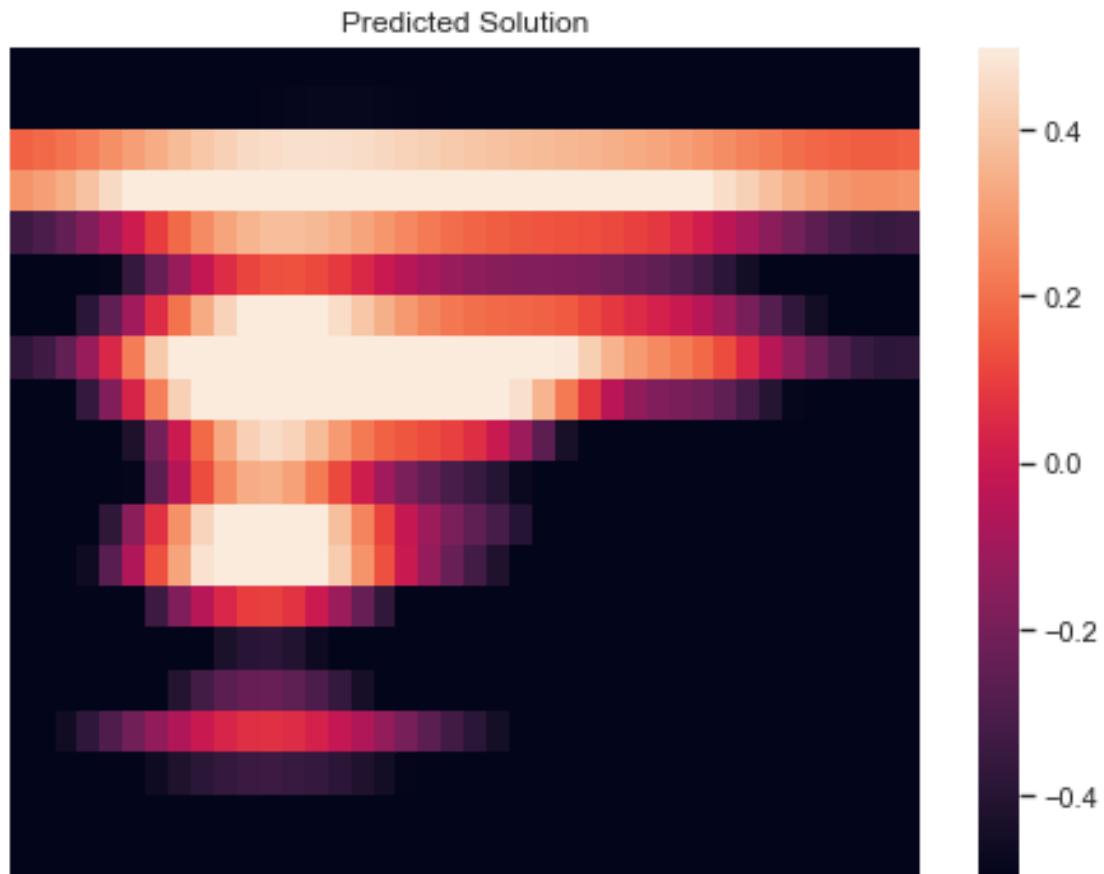
iteration 2500: loss = 0.061658553779125214, pde loss = 0.061651911586523056

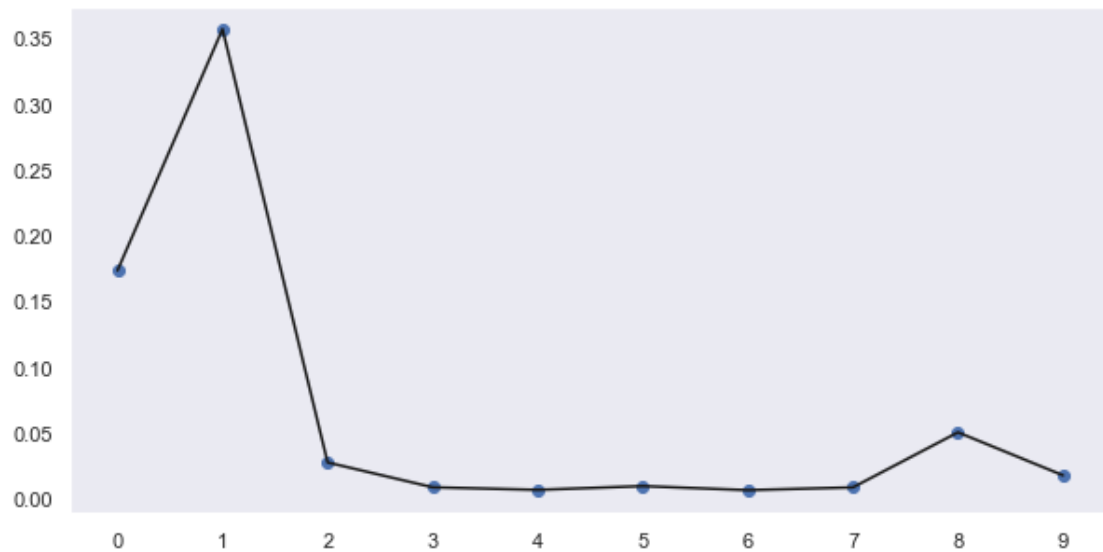
iteration 3000: loss = 0.09861291944980621, pde loss = 0.09861288964748383  
iteration 3500: loss = 0.05763258412480354, pde loss = 0.057628270238637924  
iteration 4000: loss = 0.041846565902233124, pde loss = 0.041846200823783875



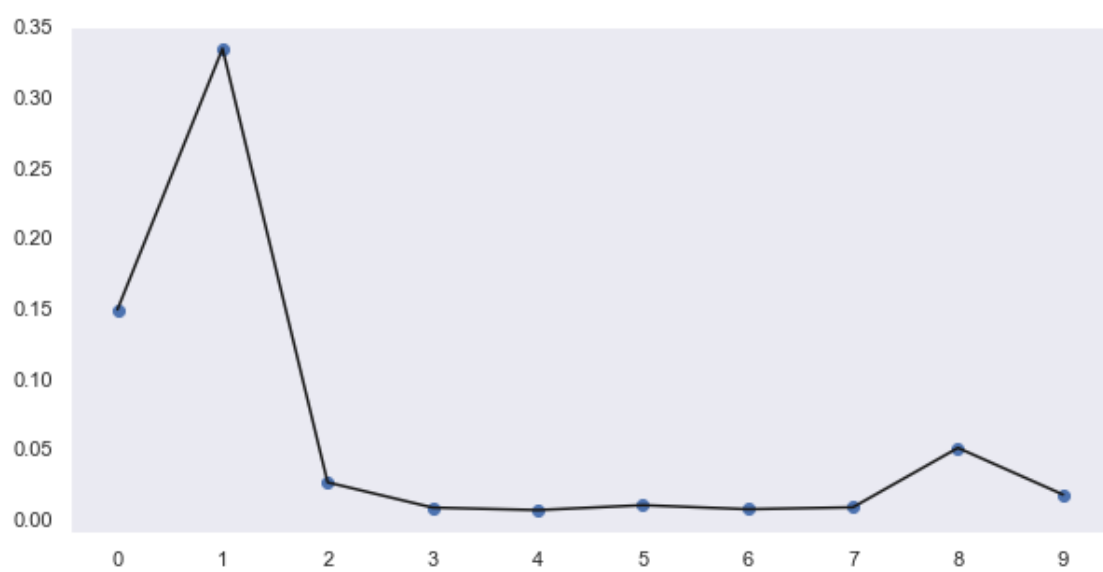
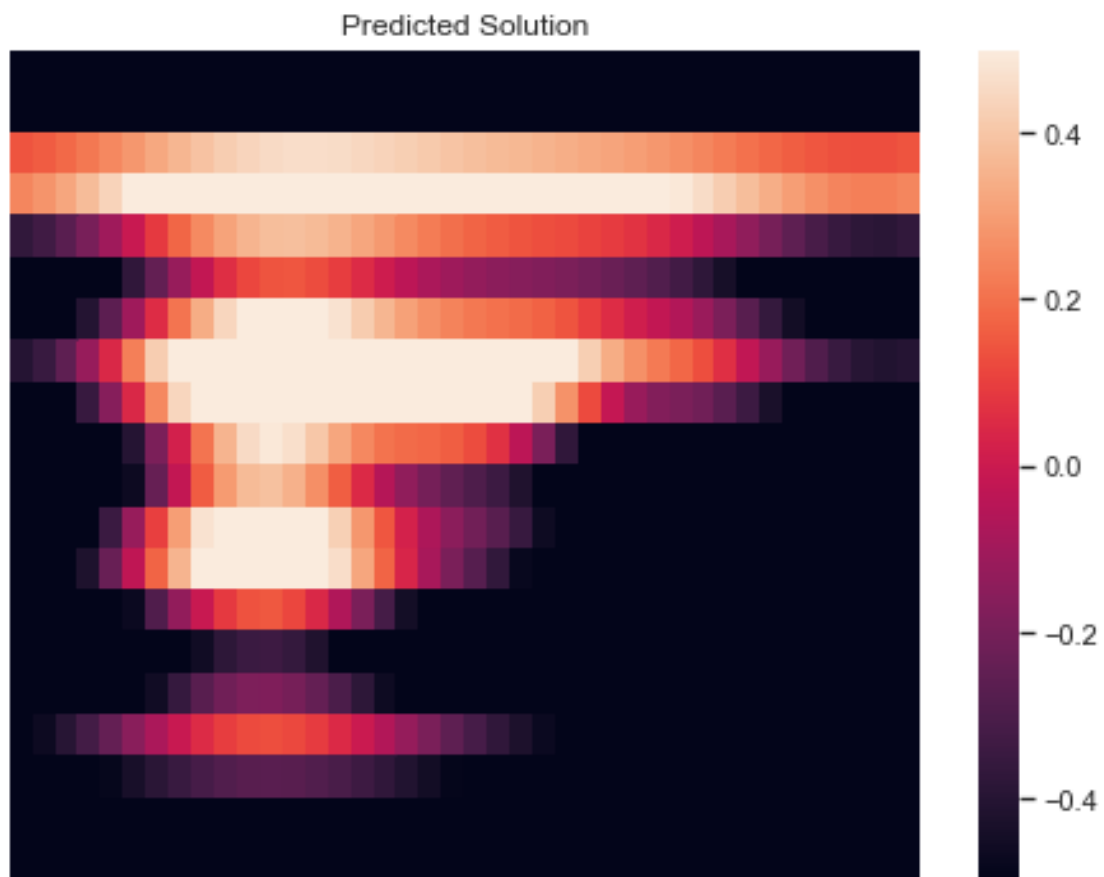


```
iteration 4500: loss = 0.06343748420476913, pde loss = 0.06343592703342438
iteration 5000: loss = 0.04104648903012276, pde loss = 0.04103073105216026
iteration 5500: loss = 0.07430292665958405, pde loss = 0.07430242002010345
iteration 6000: loss = 0.0686970129609108, pde loss = 0.06869211792945862
```

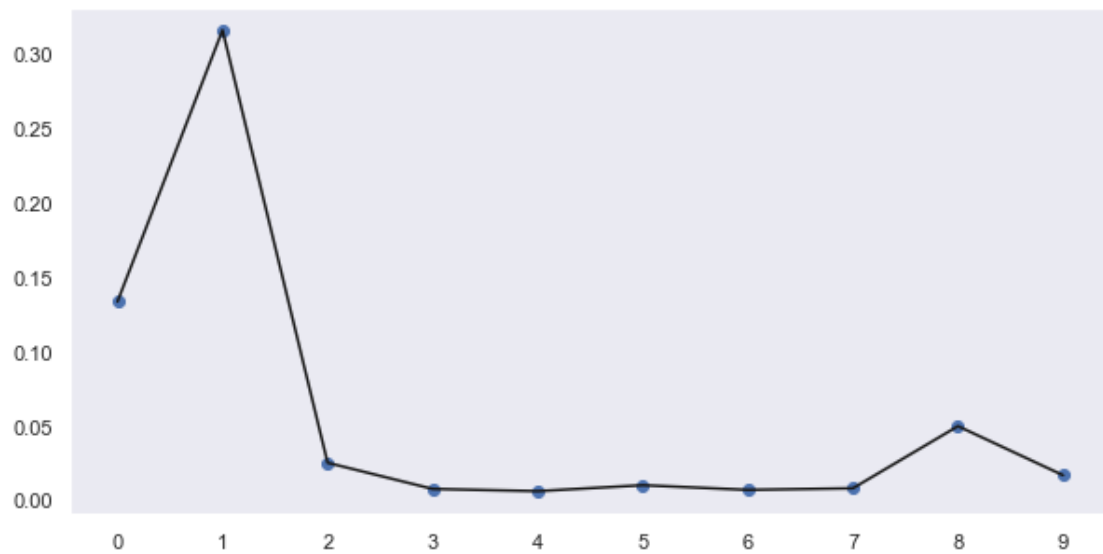
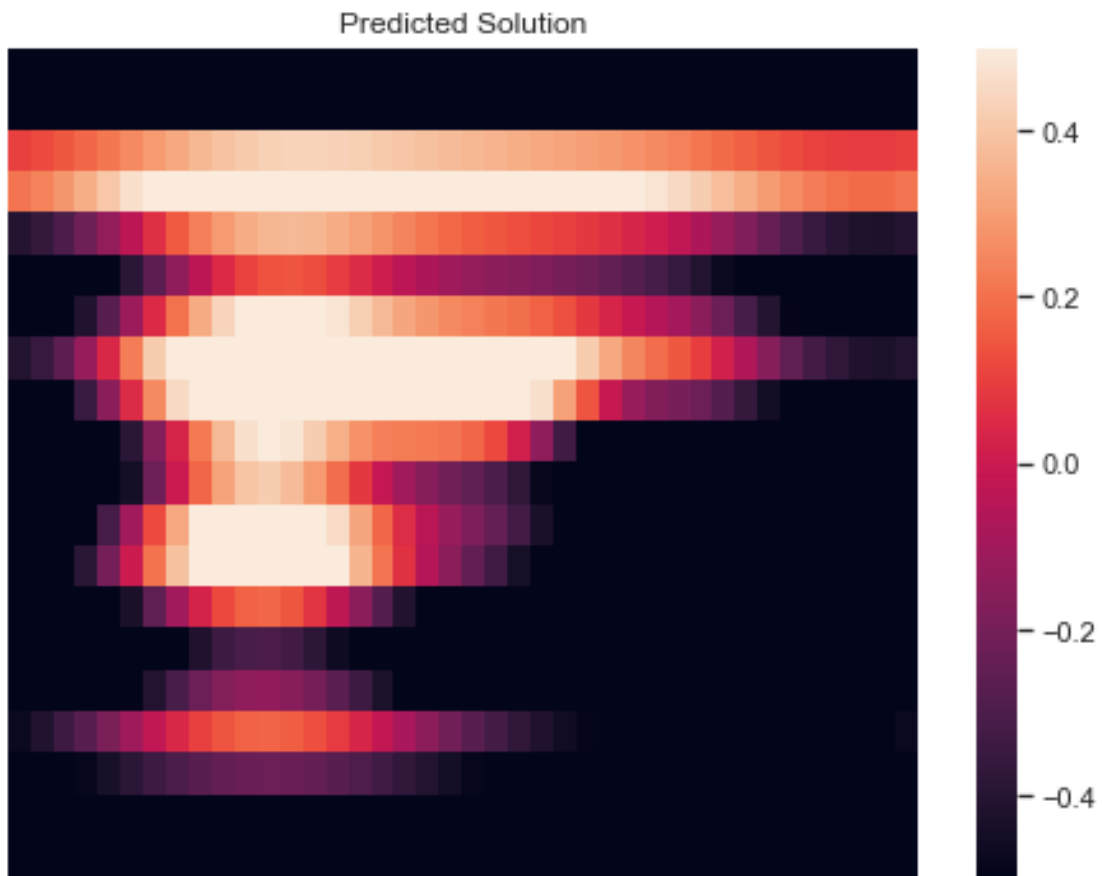




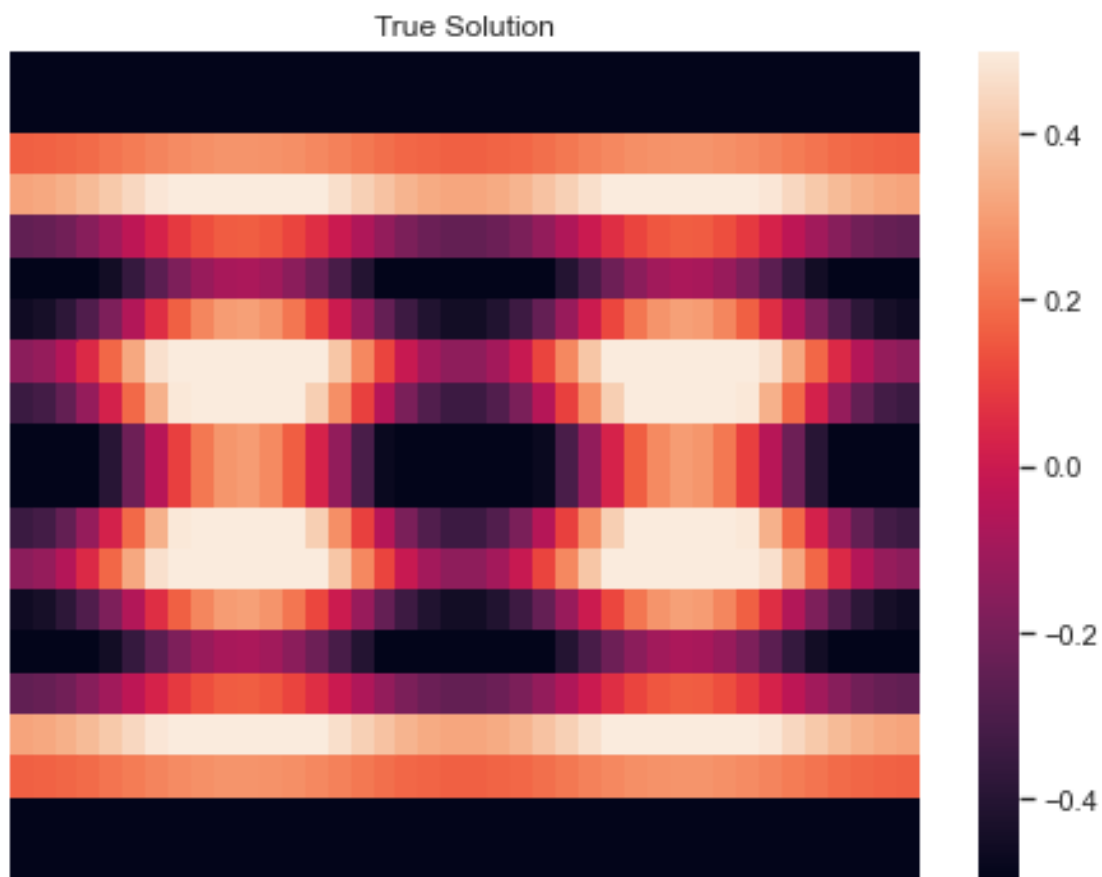
```
iteration 6500: loss = 0.03229404240846634, pde loss = 0.03227624297142029
iteration 7000: loss = 0.038880154490470886, pde loss = 0.03887774050235748
iteration 7500: loss = 0.033126018941402435, pde loss = 0.03312535211443901
iteration 8000: loss = 0.035548772662878036, pde loss = 0.03554203361272812
```

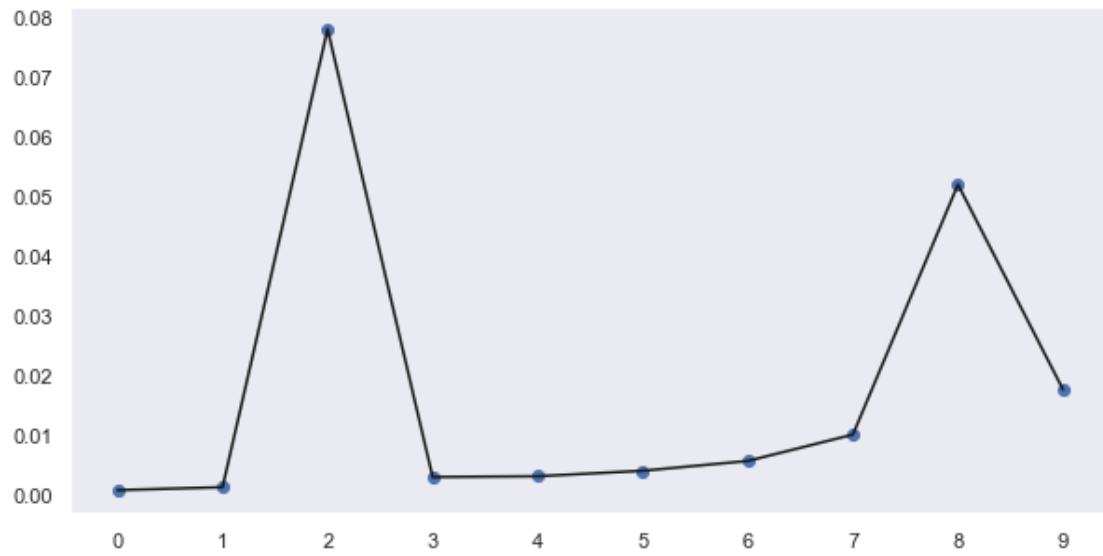


iteration 8500: loss = 0.04236430674791336, pde loss = 0.04235455393791199  
iteration 9000: loss = 0.029582271352410316, pde loss = 0.029579197987914085  
iteration 9500: loss = 0.03298865258693695, pde loss = 0.032981209456920624

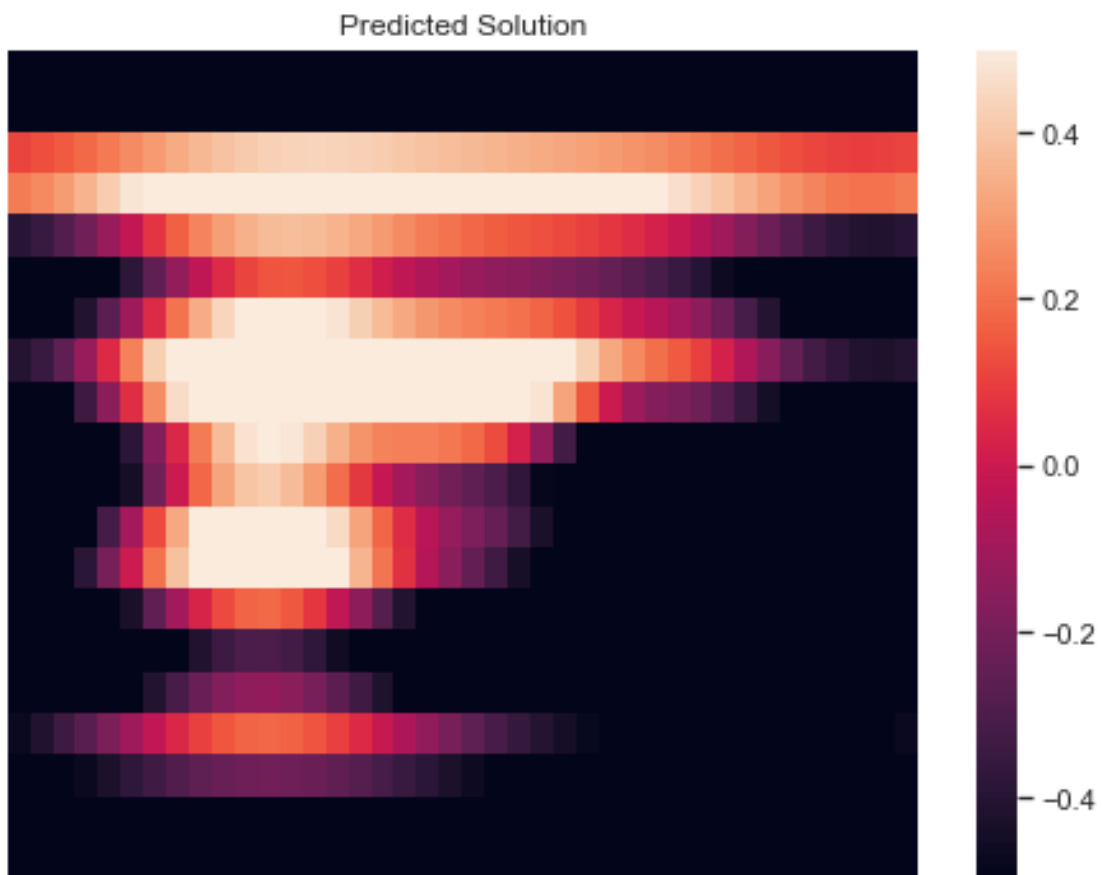


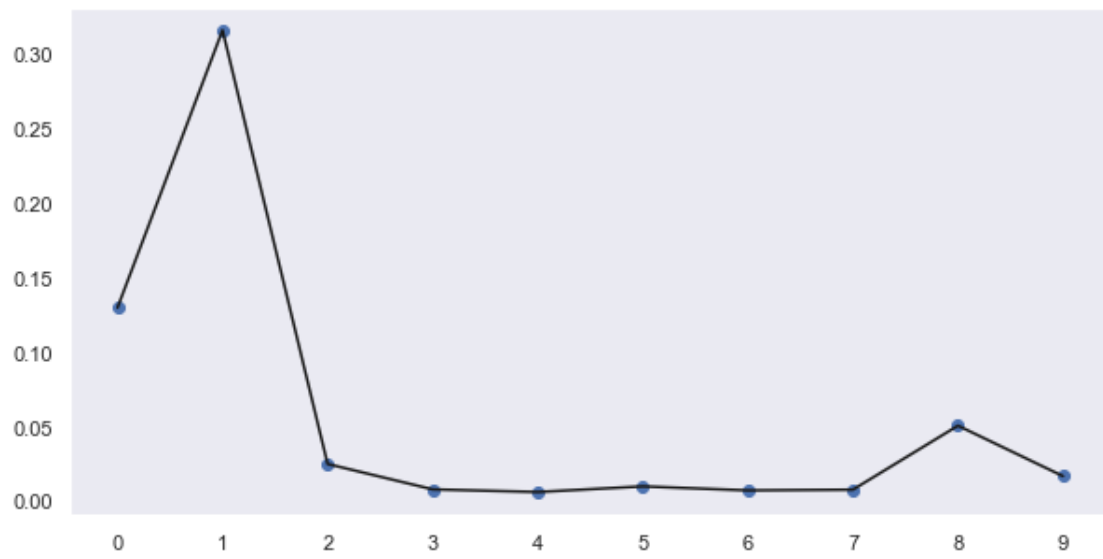
```
[ ]: model.train()
```





iteration 0: loss = 0.05028931051492691, pde loss = 0.050275832414627075





```
[ ]: test
```