In [1]:
```
import pymc3 as pm
import numpy as np
import pandas as pd
from theano import shared
import scipy.stats as stats
import matplotlib.pyplot as plt
import arviz as az
```

```
                anaconda3\envs\pm3env\lib\site-packages\scipy\__init__.py:146: UserWa
rning: A NumPy version >=1.16.5 and <1.23.0 is required for this version of SciPy (d
etected version 1.23.5
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion}"
WARNING (theano.tensor.blas): Using NumPy C-API based implementation for BLAS functi
ons.
```

In [2]:
```
az.style.use('arviz-darkgrid')
```

In [3]:
```
# DATASET_URL = "https://gist.github.com/BirajCoder/5f068dfe759c1ea6bdfce9535acdb72
DATA_FILENAME = "insurance.csv"
# download_url(DATASET_URL, '.')
```

In [4]:
```
dataframe_raw = pd.read_csv(DATA_FILENAME)
dataframe_raw.head()
```

Out[4]:

	age	sex	bmi	children	smoker	region	charges
0	19	female	27.900	0	yes	southwest	16884.92400
1	18	male	33.770	1	no	southeast	1725.55230
2	28	male	33.000	3	no	southeast	4449.46200
3	33	male	22.705	0	no	northwest	21984.47061
4	32	male	28.880	0	no	northwest	3866.85520

I choose "age" and "bmi" as the independent variables, because they are numerical variables.

I choose "charges" as dependent variables.

My goal is to find the linear relationship among these three variables; and hopefully, use the data of "age" and "bmi" to predict the "charges".

In [5]:
```
not_smoker = dataframe_raw[dataframe_raw.smoker=='no']
```

For now, I am focusing on people who are not smokers.

In [6]:
```
nsmokerframe = not_smoker[['age','bmi','charges']]
```

In [7]:
```
nsmokerframe['age_c'] = nsmokerframe.age - np.mean(nsmokerframe.age)
```

I centered the independent variables, same with "bmi".

```
                    \Local\Temp\ipykernel_23924\2369034674.py:1: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/u
ser_guide/indexing.html#returning-a-view-versus-a-copy
  nsmokerframe['age_c'] = nsmokerframe.age - np.mean(nsmokerframe.age)
```

In [8]:
```
nsmokerframe['bmi_c'] = nsmokerframe.bmi - np.mean(nsmokerframe.bmi)
```

```
                    \Local\Temp\ipykernel_23924\2603070654.py:1: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/u
ser_guide/indexing.html#returning-a-view-versus-a-copy
  nsmokerframe['bmi_c'] = nsmokerframe.bmi - np.mean(nsmokerframe.bmi)
```

In [9]:
```
nsmokerframe['charge_thousand'] = nsmokerframe.charges/1000
```

```
                    \Local\Temp\ipykernel_23924\3710568513.py:1: SettingWithCopyWa
rning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pan
ser_guide/indexing.html#returning-a-view-versus-a-copy
  nsmokerframe['charge_thousand'] = nsmokerframe.charges/1000
```

I changed "charge" to "charge_thousand" just want to have smaller numbers in my outcome.

In [10]:
```
x_mean= [np.mean(nsmokerframe.age),np.mean(nsmokerframe.bmi)]
np.asarray(x_mean)
```

Out[10]:
```
array([39.38533835, 30.65179511])
```

In [11]:
```
in_array = nsmokerframe[['age_c','bmi_c']].to_numpy()
out_array = nsmokerframe['charge_thousand'].to_numpy()
```

In [25]:
```
with pm.Model() as model_naive:
    α_tmp = pm.Normal('α_tmp', mu=0, sd=10)
    β = pm.Normal('β', mu=0, sd=1, shape=2)
    ϵ = pm.HalfCauchy('ϵ', 5)

    μ = α_tmp + pm.math.dot(in_array, β)

    α = pm.Deterministic('α', α_tmp - pm.math.dot(x_mean, β))

    y_pred = pm.Normal('y_pred', mu=μ, sd=ϵ, observed=out_array)

    trace_naive = pm.sample(1000)
```

I applied a very simple model arbitrarily to see what the outcome will look like, so that I can improve from there.

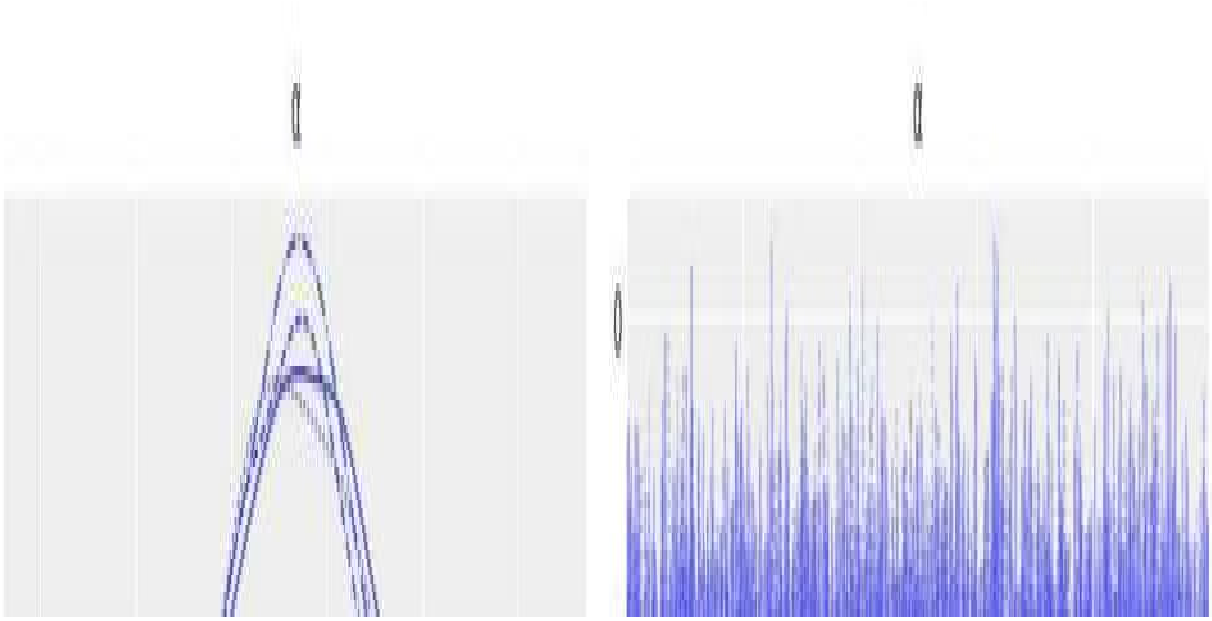
Following are the posterior distributions and summaries I got from the model.

In [19]:
```
varnames = ['α', 'β', 'ϵ']
az.plot_trace(trace_naive, var_names=varnames);
az.summary(trace_naive, var_names=varnames)
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
             \anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
Got error No model on context stack. trying to find log_likelihood in translation.
Got error No model on context stack. trying to find log_likelihood in translation.
C:\Users\Mengj\anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
```

Out[19]:

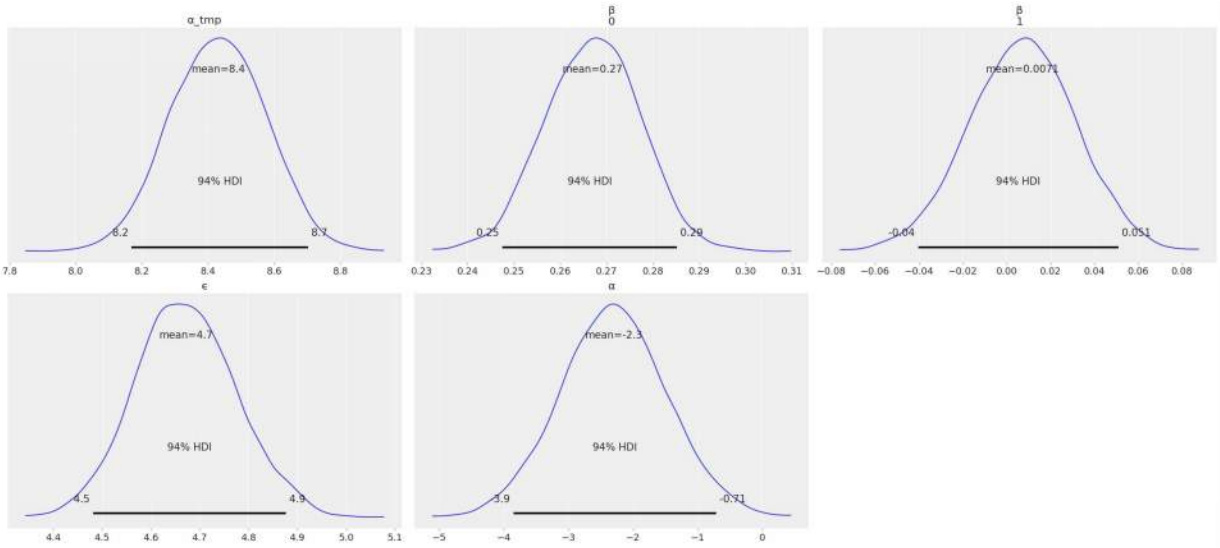
	mean	sd	hdi_3%	hdi_97%	mcse_mean	mcse_sd	ess_bulk	ess_tail	r_hat
α	-2.301	0.828	-3.854	-0.713	0.011	0.008	6137.0	3044.0	1.00
β[0]	0.267	0.010	0.247	0.285	0.000	0.000	6005.0	3182.0	1.00
β[1]	0.007	0.024	-0.040	0.051	0.000	0.000	5473.0	3284.0	1.01
ϵ	4.672	0.104	4.482	4.876	0.001	0.001	6617.0	3031.0	1.00



In [20]:
```
az.plot_posterior(trace_naive)
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
            \anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
```

Out[20]:
```
array([[<Axes: title={'center': 'α_tmp'}>,
        <Axes: title={'center': 'β\n0'}>,
        <Axes: title={'center': 'β\n1'}>],
       [<Axes: title={'center': 'ϵ'}>, <Axes: title={'center': 'α'}>,
        <Axes: >]], dtype=object)
```



In [26]:
```
ppc = pm.sample_posterior_predictive(trace_naive, samples=200, model=model_naive, r
```

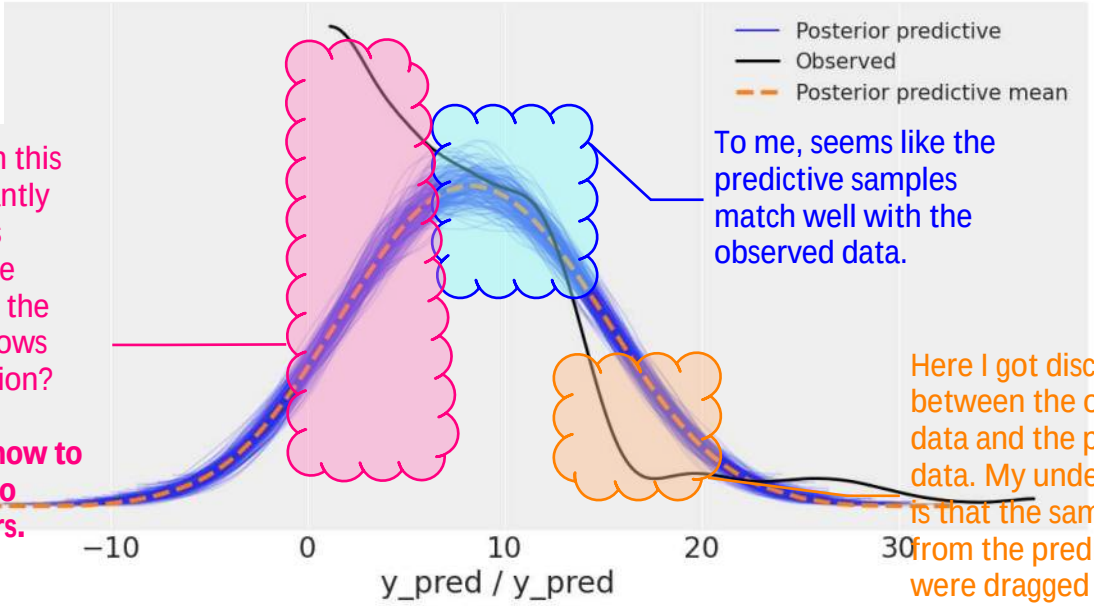
```
            \anaconda3\envs\pm3env\lib\site-packages\pymc3\sampling.py:1708: UserW
arning: samples parameter is smaller than nchains times ndraws, some draws and/or ch
ains may not be represented in the returned posterior predictive sample
  warnings.warn(
```

100.00% [200/200 00:00<00:00]

In [27]:
```
data_ppc = az.from_pymc3(trace=trace_naive, posterior_predictive=ppc)
ax = az.plot_ppc(data_ppc, figsize=(12, 6), mean=True)
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
             \anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
posterior predictive variable y_pred's shape not compatible with number of chains an
d draws. This can mean that some draws or even whole chains are not represented.
```

Here is the posterior predictive samples I got from my first naive model.



The prediction in this range is significantly off. I think this is resulted from the assumption that the "y(charges)" follows normal distribution?

My question is how to restrict my "y" to positive numbers.

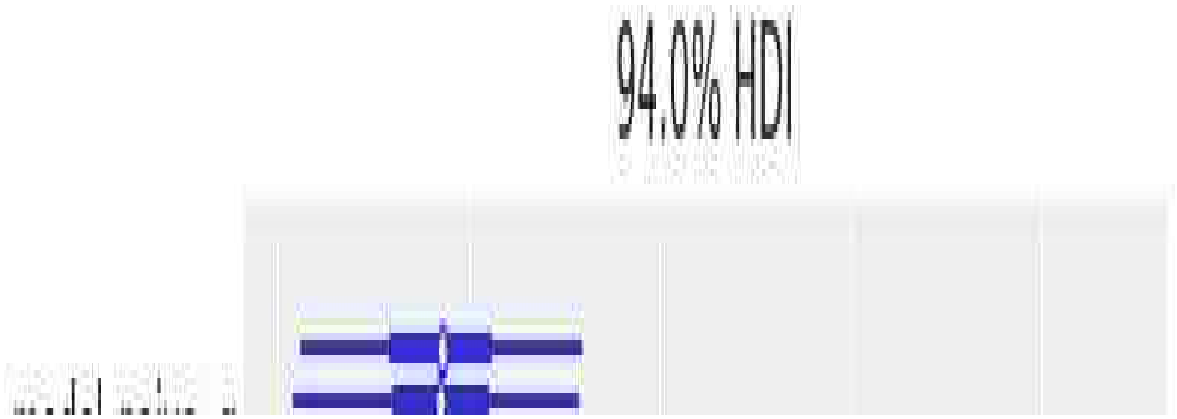
To me, seems like the predictive samples match well with the observed data.

Here I got discrepancies between the observed data and the predicted data. My understanding is that the samples from the prediction were dragged towards the outliers.

In [13]:
```
az.plot_forest([trace_naive],
               model_names=['model_naive'],
               var_names=['α', 'β', 'ϵ'],
               combined=False, colors='cycle', figsize=(8, 3))
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
             \anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
```

Out[13]:
```
array([<Axes: title={'center': '94.0% HDI'}>], dtype=object)
```

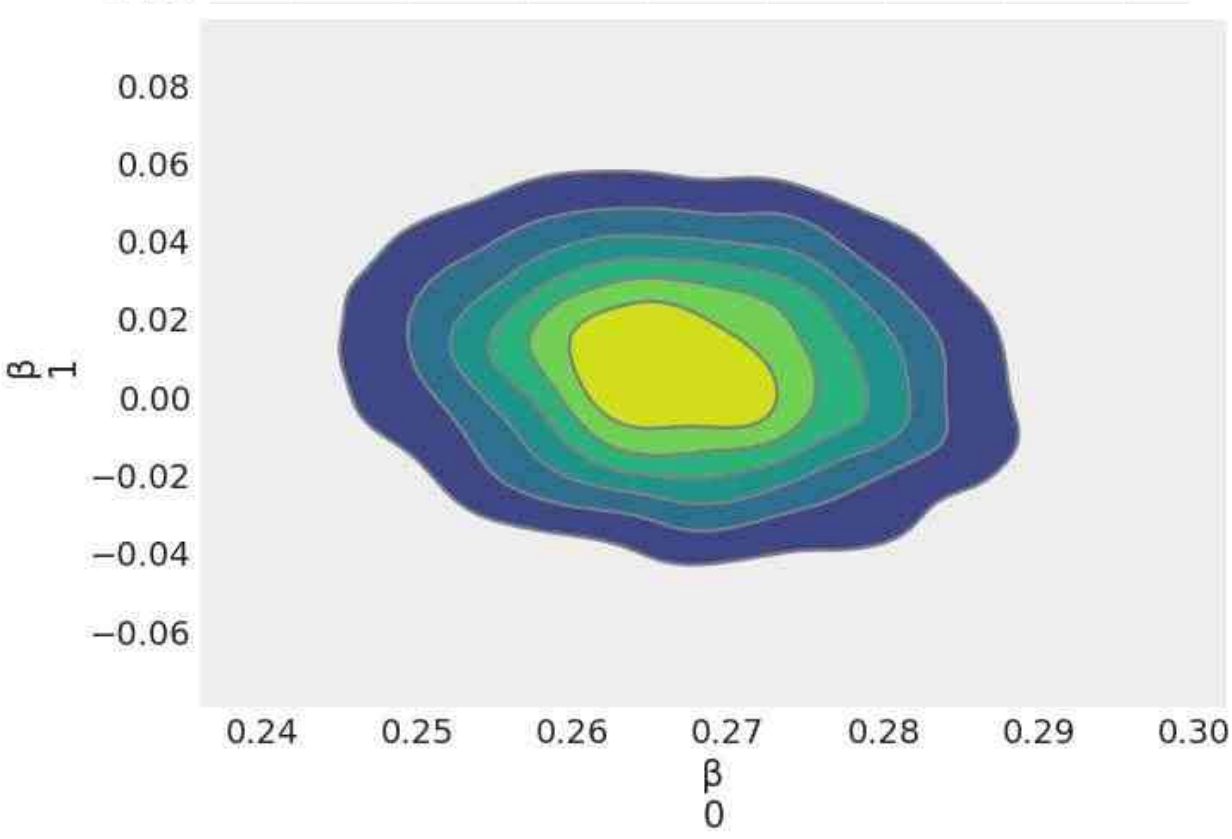


I also did a quick plot to have a glance at the correlation between "age" and "bmi".

In [14]:
```
az.plot_pair(trace_naive, var_names=['β']
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
             \anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
```

Out[14]: <Axes: xlabel='β\n0', ylabel='β\n1'>



I calculated the covariance and correlation between 2 independent variables. It seems like they are not highly correlated

In [15]:
```
covariance = np.cov(nsmokerframe['age_c'], nsmokerframe['bmi_c'])
covariance
```

Out[15]:
```
array([[198.3424377 ,  10.43742616],
       [ 10.43742616,  36.51919527]])
```

In [18]:
```
corr,_ = stats.pearsonr(nsmokerframe['age_c'], nsmokerframe['bmi_c'])
corr
```

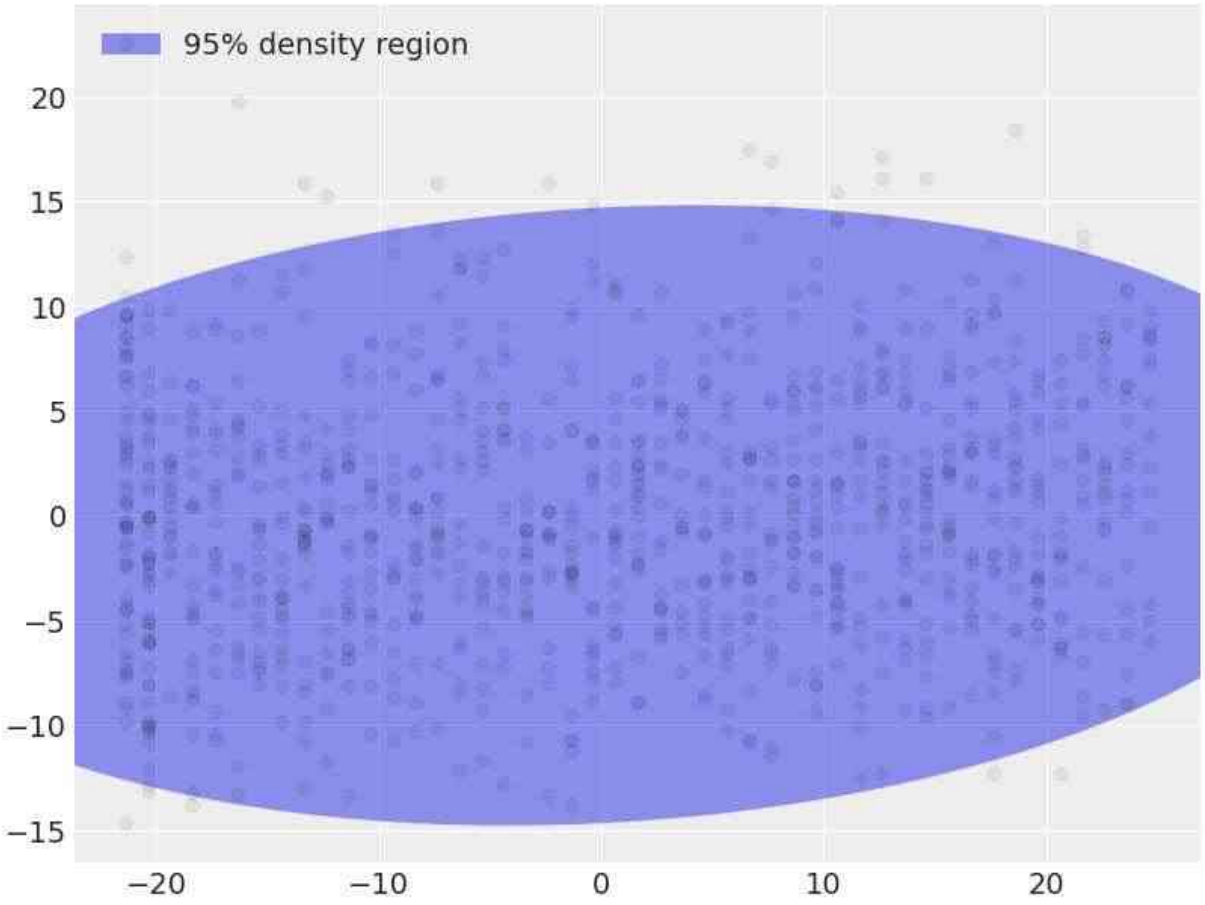
Out[18]: 0.12263798130263143

I borrowed the plotting code from the example of "LKJ Cholesky Covariance Prior for Multivariate Normal Models" and plotted these 2 variables -- (x = age_c, y= bmi_c)

```
import seaborn as sns
from matplotlib.patches import Ellipse
```

```
var, U = np.linalg.eig(covariance)
angle = 180.0 / np.pi * np.arccos(np.abs(U[0, 0]))

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

blue, _, red, *_ = sns.color_palette()

e = Ellipse(μ_actual, 2 * np.sqrt(5.991 * var[0]), 2 * np.sqrt(5.991 * var[1]), ang
e.set_alpha(0.5)
e.set_facecolor(blue)
e.set_zorder(10)
ax.add_artist(e)

ax.scatter(nsmokerframe['age_c'], nsmokerframe['bmi_c'], c="k", alpha=0.05, zorder=
```

```
rect = plt.Rectangle((0, 0), 1, 1, fc=blue, alpha=0.5)
ax.legend([rect], ["95% density region"], loc=2);
```



```
            \anaconda3\envs\pm3env\lib\site-packages\deprecat\classic.py:215: Futu
reWarning: In v4.0, pm.sample will return an `arviz.InferenceData` object instead of
a `MultiTrace` by default. You can pass return_inferencedata=True or return_inferenc
edata=False to be safe and silence this warning.
  return wrapped_(*args_, **kwargs_)
Auto-assigning NUTS sampler...
Initializing NUTS using jitter+adapt_diag...
Multiprocess sampling (4 chains in 4 jobs)
NUTS: [ϵ, β, α_tmp]
```

Messages while running the model_naive

100.00% [8000/8000 00:18<00:00 Sampling 4 chains, 0 divergences]

```
Sampling 4 chains for 1_000 tune and 1_000 draw iterations (4_000 + 4_000 draws tota
l) took 43 seconds.
```

In [12]:

```
with pm.Model() as m_x1x2:
    α = pm.Normal('α', mu=0, sd=10)
    β1 = pm.Normal('β1', mu=0, sd=10)
    β2 = pm.Normal('β2', mu=0, sd=10)
    ϵ = pm.HalfCauchy('ϵ', 5)

    μ = α + β1 * in_array[:, 0] + β2 * in_array[:, 1]

    y_pred = pm.Normal('y_pred', mu=μ, sd=ϵ, observed=out_array)

    trace_x1x2 = pm.sample(1000)


with pm.Model() as m_x1:
    α = pm.Normal('α', mu=0, sd=10)
    β1 = pm.Normal('β1', mu=0, sd=10)
    ϵ = pm.HalfCauchy('ϵ', 5)

    μ = α + β1 * in_array[:, 0]

    y_pred = pm.Normal('y_pred', mu=μ, sd=ϵ, observed=out_array)

    trace_x1 = pm.sample(1000)

with pm.Model() as m_x2:
    α = pm.Normal('α', mu=0, sd=10)
    β2 = pm.Normal('β2', mu=0, sd=10)
    ϵ = pm.HalfCauchy('ϵ', 5)

    μ = α + β2 * in_array[:, 1]

    y_pred = pm.Normal('y_pred', mu=μ, sd=ϵ, observed=out_array)

    trace_x2 = pm.sample(1000)
```

To learn the knowledge about "confounding effects" in multi-variances linear models. I applied and compared the following three models --
i) same model as the "naive one" with both "age" and "bmi" as independent variables.

ii) a simple linear regression model with only "age(x1)" as the independent variable.

iii) a simple linear regression model with only "bmi(x2)" as the independent variable.

```
            \anaconda3\envs\pm3env\lib\site-packages\deprecat\classic.py:215: Futu
rewarning: in v4.0, pm.sample will return an `arviz.InferenceData` object instead of
a `MultiTrace` by default. You can pass return_inferencedata=True or return_inferenc
edata=False to be safe and silence this warning.
  return wrapped_(*args_, **kwargs_)
Auto-assigning NUTS sampler...
Initializing NUTS using jitter+adapt_diag...
Multiprocess sampling (4 chains in 4 jobs)
NUTS: [ϵ, β2, β1, α]
```

```
100.00% [8000/8000 00:16<00:00 Sampling 4 chains, 0 divergences]
```

```
Sampling 4 chains for 1_000 tune and 1_000 draw iterations (4_000 + 4_000 draws tota
l) took 41 seconds.
            \anaconda3\envs\pm3env\lib\site-packages\deprecat\classic.py:215: Futu
reWarning: In v4.0, pm.sample will return an `arviz.InferenceData` object instead of
a `MultiTrace` by default. You can pass return_inferencedata=True or return_inferenc
edata=False to be safe and silence this warning.
  return wrapped_(*args_, **kwargs_)
Auto-assigning NUTS sampler...
Initializing NUTS using jitter+adapt_diag...
Multiprocess sampling (4 chains in 4 jobs)
NUTS: [ϵ, β1, α]
```

```
100.00% [8000/8000 00:16<00:00 Sampling 4 chains, 0 divergences]
```

```
Sampling 4 chains for 1_000 tune and 1_000 draw iterations (4_000 + 4_000 draws tota
l) took 39 seconds.
            \anaconda3\envs\pm3env\lib\site-packages\deprecat\classic.py:215: Futu
rewarning: in v4.0, pm.sample will return an `arviz.InferenceData` object instead of
a `MultiTrace` by default. You can pass return_inferencedata=True or return_inferenc
edata=False to be safe and silence this warning.
  return wrapped_(*args_, **kwargs_)
Auto-assigning NUTS sampler...
Initializing NUTS using jitter+adapt_diag...
Multiprocess sampling (4 chains in 4 jobs)
NUTS: [ϵ, β2, α]
```

```
100.00% [8000/8000 00:15<00:00 Sampling 4 chains, 0 divergences]
```

```
Sampling 4 chains for 1_000 tune and 1_000 draw iterations (4_000 + 4_000 draws tota
l) took 39 seconds.
```

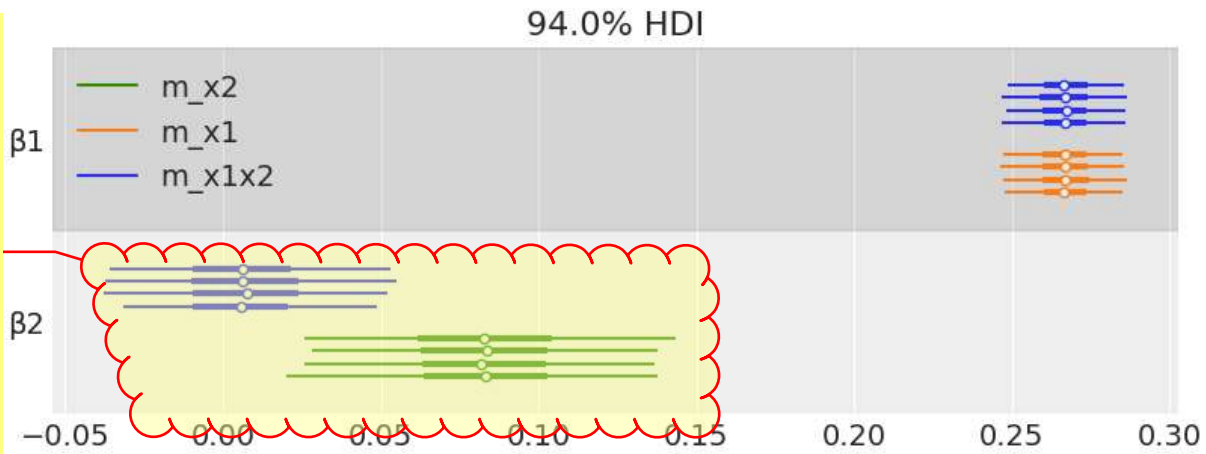
In [13]:
```
az.plot_forest([trace_x1x2, trace_x1, trace_x2],
               model_names=['m_x1x2', 'm_x1', 'm_x2'],
               var_names=['β1', 'β2'],
               combined=False, colors='cycle', figsize=(8, 3))
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
              \anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
Got error No model on context stack. trying to find log_likelihood in translation.
Got error No model on context stack. trying to find log_likelihood in translation.
```

Out[13]: array([<Axes: title={'center': '94.0% HDI'}>], dtype=object)

When "bmi" is the only variable, it will have a lager influence on "charge"; while the influence of "age" is pretty much the same in both models.

I can describe the result, but I am not sure how to interpret it. What is this difference indicating?



In [28]:
```
with pm.Model() as model_tnaive:
    α_tmp = pm.Normal('α_tmp', mu=0, sd=10)
    β = pm.Normal('β', mu=0, sd=10, shape=2)
    ϵ = pm.HalfCauchy('ϵ', 5)
    ν_ = pm.Exponential('ν_', 1/15)
    ν = pm.Deterministic('ν', ν_ + 1)

    μ = α_tmp + pm.math.dot(in_array, β)

    α = pm.Deterministic('α', α_tmp - pm.math.dot(x_mean, β))

    y_pred = pm.StudentT('y_pred', mu=μ,
                         sd=ϵ, nu=ν, observed=out_array)


    trace_tnaive = pm.sample(1000)
```

From the first naive model, I noticed that the posterior predictions of charges were dragged towards the outliers in the observed data. I thought it could be helpful to assume "y(charges)" as Student-T distribution.

Self Note -- What is the difference between HalfCauchy and HalfNormal?

```
              \anaconda3\envs\pm3env\lib\site-packages\deprecat\classic.py:215: Futu
reWarning: In v4.0, pm.sample will return an `arviz.InferenceData` object instead of
a `MultiTrace` by default. You can pass return_inferencedata=True or return_inferenc
edata=False to be safe and silence this warning.
  return wrapped_(*args_, **kwargs_)
Auto-assigning NUTS sampler...
Initializing NUTS using jitter+adapt_diag...
Multiprocess sampling (4 chains in 4 jobs)
NUTS: [ν_, ϵ, β, α_tmp]
```


100.00% [8000/8000 00:22<00:00 Sampling 4 chains, 27 divergences]

```
Sampling 4 chains for 1_000 tune and 1_000 draw iterations (4_000 + 4_000 draws tota
l) took 46 seconds.
There were 2 divergences after tuning. Increase `target_accept` or reparameterize.
There were 9 divergences after tuning. Increase `target_accept` or reparameterize.
There were 9 divergences after tuning. Increase `target_accept` or reparameterize.
There were 7 divergences after tuning. Increase `target_accept` or reparameterize.
```

I got these warning messages after sampling. Can I ask what they are indicating?

But then when I looked at the posterior summaries, and especially the posterior predictions, very funny results showed up...

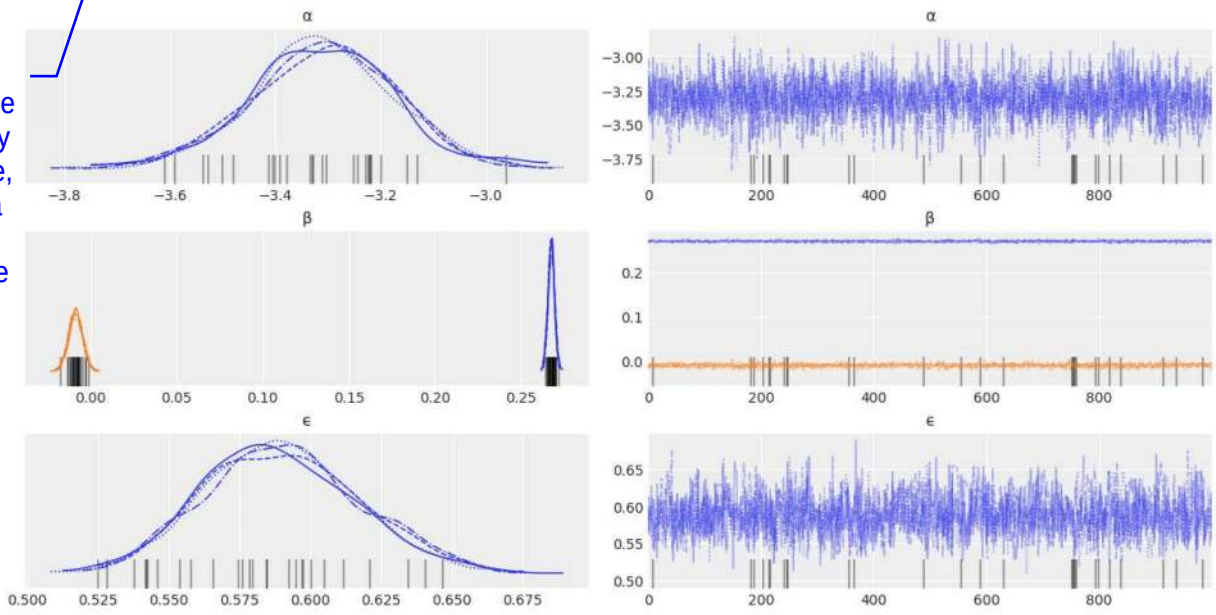
In [29]:
```
varnames = ['α', 'β', 'ϵ']
az.plot_trace(trace_tnaive, var_names=varnames);
az.summary(trace_tnaive, var_names=varnames)
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
              \anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
Got error No model on context stack. trying to find log_likelihood in translation.
Got error No model on context stack. trying to find log_likelihood in translation.
C:\Users\Mengj\anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
```

Out[29]:

	mean	sd	hdi_3%	hdi_97%	mcse_mean	mcse_sd	ess_bulk	ess_tail	r_hat
α	-3.309	0.139	-3.600	-3.073	0.003	0.002	2340.0	1822.0	1.0
β[0]	0.268	0.002	0.264	0.271	0.000	0.000	1914.0	2036.0	1.0
β[1]	-0.009	0.004	-0.017	-0.002	0.000	0.000	2143.0	2176.0	1.0
ϵ	0.590	0.027	0.538	0.638	0.001	0.001	1518.0	1196.0	1.0

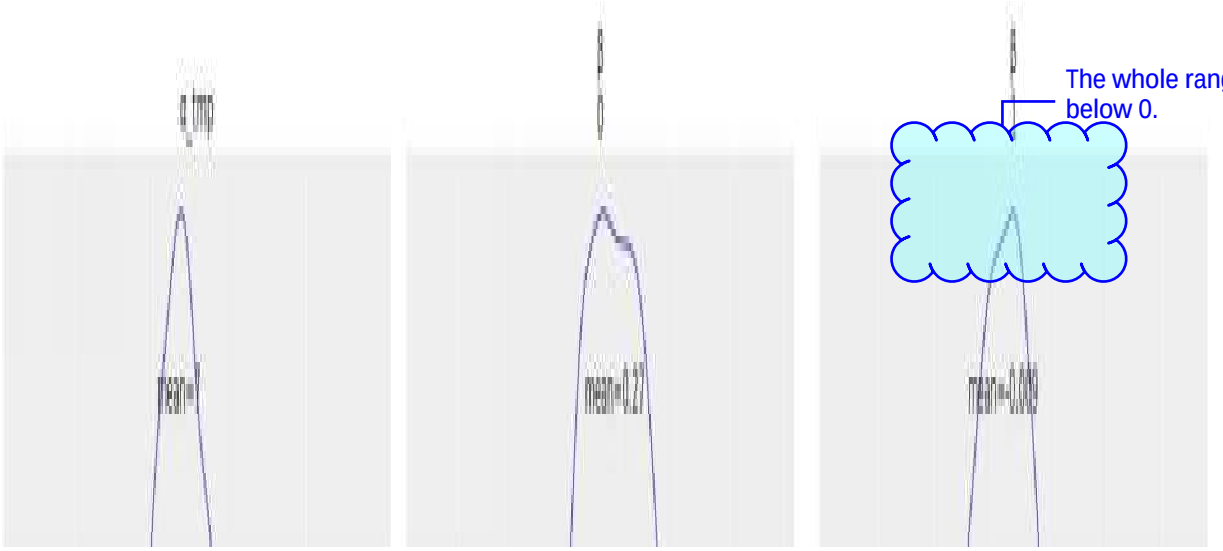
Though still very close to 0, the beta related to "bmi" turns to 0. This does not make much sense to me, because when passed a certain range, the higher the bmi, the less healthy the person might be, which may lead to a higher expense on the health insurance charge.



In [16]:
```
az.plot_posterior(trace_tnaive)
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
              \anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
```

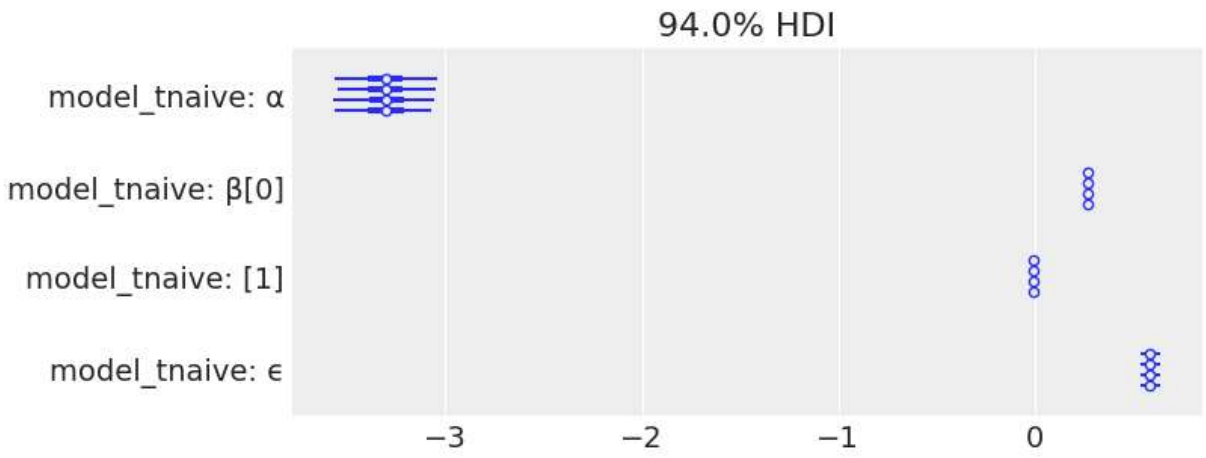
Out[16]:
```
array([[<Axes: title={'center': 'α_tmp'}>,
        <Axes: title={'center': 'β\n0'}>,
        <Axes: title={'center': 'β\n1'}>],
       [<Axes: title={'center': 'ϵ'}>, <Axes: title={'center': 'v'}>,
        <Axes: title={'center': 'α'}>]], dtype=object)
```



In [17]:
```
az.plot_forest([trace_tnaive],
               model_names=['model_tnaive'],
               var_names=['α', 'β', 'ϵ'],
               combined=False, colors='cycle', figsize=(8, 3))
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
```

Out[17]:
```
array([<Axes: title={'center': '94.0% HDI'}>], dtype=object)
```



In [30]:
```
ppc_t = pm.sample_posterior_predictive(trace_tnaive, samples=200, model=model_tnaiv
```

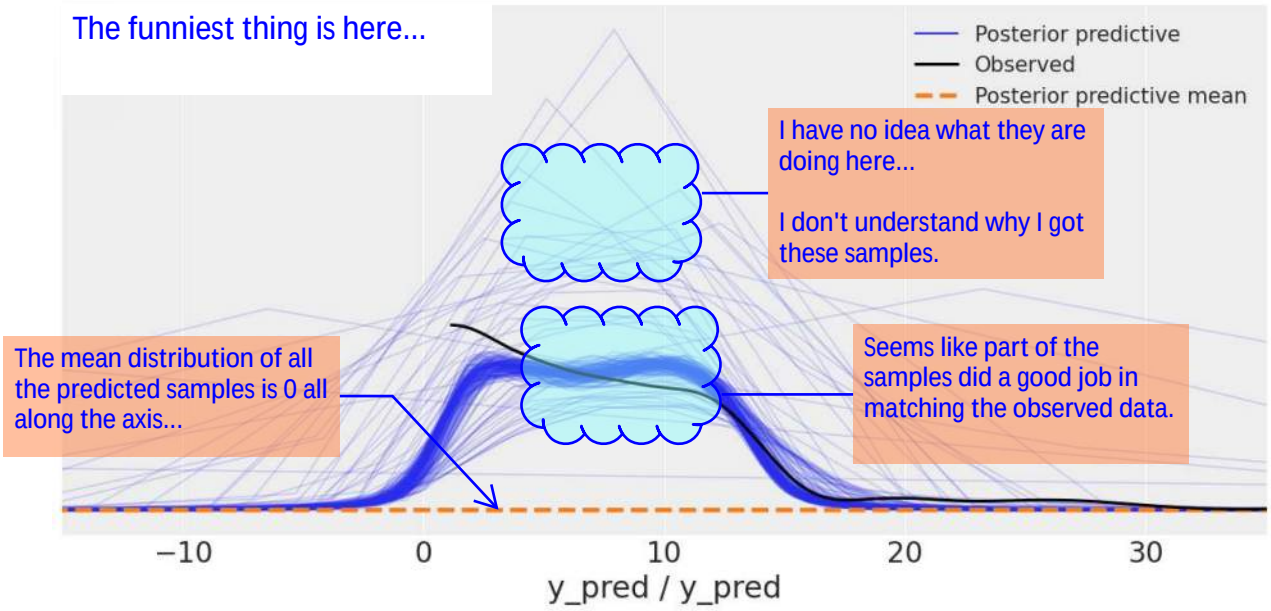
```
            \anaconda3\envs\pm3env\lib\site-packages\pymc3\sampling.py:1708: UserW
arning: samples parameter is smaller than nchains times ndraws, some draws and/or ch
ains may not be represented in the returned posterior predictive sample
  warnings.warn(
```

100.00% [200/200 00:01<00:00]

In [31]:
```
data_ppc_t = az.from_pymc3(trace=trace_tnaive, posterior_predictive=ppc_t)
ax = az.plot_ppc(data_ppc_t, figsize=(12, 6), mean=True)
plt.xlim(-15, 35)
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
              \anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
posterior predictive variable y_pred's shape not compatible with number of chains an
d draws. This can mean that some draws or even whole chains are not represented.
```

Out[31]: (-15.0, 35.0)



In [19]:
```
varnames = ['α', 'β', 'ϵ']
az.plot_trace(trace_naive, var_names=varnames);
az.summary(trace_naive, var_names=varnames)
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
              \anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
Got error No model on context stack. trying to find log_likelihood in translation.
Got error No model on context stack. trying to find log_likelihood in translation.
C:\Users\Mengj\anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
```

Out[19]:

	mean	sd	hdi_3%	hdi_97%	mcse_mean	mcse_sd	ess_bulk	ess_tail	r_hat
α	-2.301	0.828	-3.854	-0.713	0.011	0.008	6137.0	3044.0	1.00
β[0]	0.267	0.010	0.247	0.285	0.000	0.000	6005.0	3182.0	1.00
β[1]	0.007	0.024	-0.040	0.051	0.000	0.000	5473.0	3284.0	1.01
ϵ	4.672	0.104	4.482	4.876	0.001	0.001	6617.0	3031.0	1.00

In [17]:
```
with pm.Model() as model_tnaivev2:
    α_tmp = pm.StudentT('α_tmp', mu=0,sd=10, nu=30/29)
    β = pm.StudentT('β', mu=0, sd=10, nu=30/29, shape=2)
    ϵ = pm.HalfCauchy('ϵ', 5)
    #v_ = pm.Exponential('v_', 1/15)
    #v = pm.Deterministic('v', v_ + 1)

    μ = α_tmp + pm.math.dot(in_array, β)

    α = pm.Deterministic('α', α_tmp - pm.math.dot(x_mean, β))

    y_pred = pm.Normal('y_pred', mu=μ, sd=ϵ, observed=out_array)


    trace_tnaivev2 = pm.sample(1000)
```

Given that very funny outcome on my 2nd trial of my model, I tried to change my priors to Student-T distribution instead.

I could be wrong, but I was hoping that putting the Student-T distribution on my priors would make them less informative, so that the discrepancy between posterior predictions and the observed data could be narrowed.

I set the freedom as 30/29 (1+1/29), which now I don't think it is necessary because it is the prior distribution.I picked the number >1 when writing this model because I learned from the book that larger freedom works better with data having more outliers...

```
              anaconda3\envs\pm3env\lib\site-packages\deprecat\cl
reWarning: In v4.0, pm.sample will return an `arviz.InferenceData`
a `MultiTrace` by default. You can pass return_inferencedata=True
edata=False to be safe and silence this warning.
  return wrapped_(*args_, **kwargs_)
Auto-assigning NUTS sampler...
Initializing NUTS using jitter+adapt_diag...
Multiprocess sampling (4 chains in 4 jobs)
NUTS: [ϵ, β, α_tmp]
```

100.00% [8000/8000 00:07<00:00 Sampling 4 chains, 0 divergences]

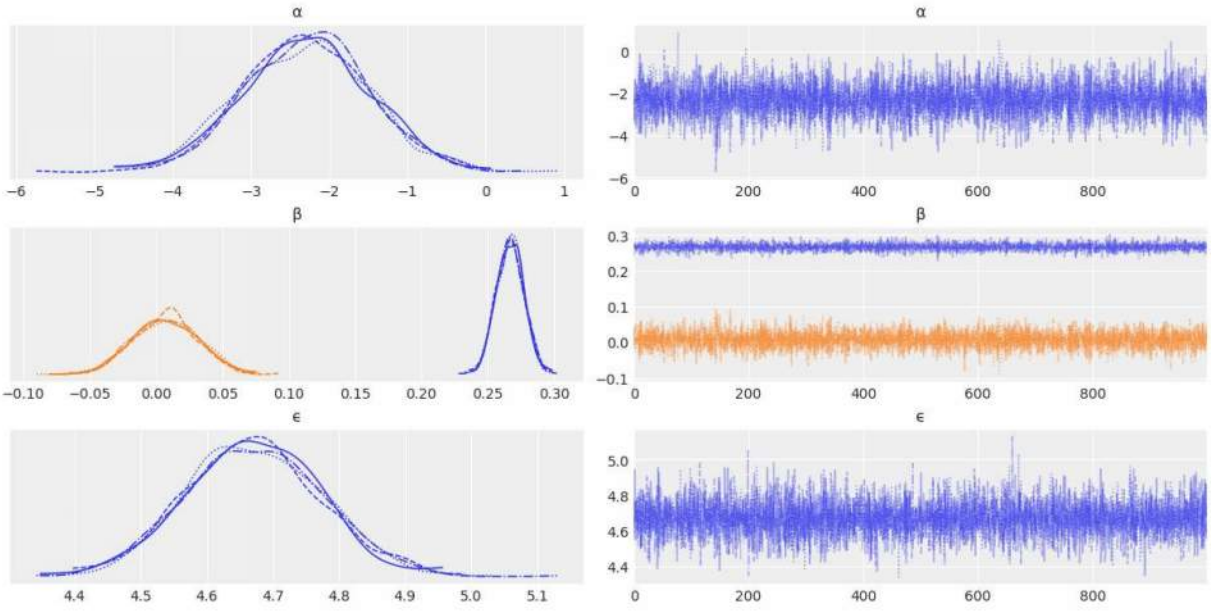
```
Sampling 4 chains for 1_000 tune and 1_000 draw iterations (4_000 + 4_000 draws tota
l) took 25 seconds.
```

In [18]:
```
varnames = ['α', 'β', 'ϵ']
az.plot_trace(trace_tnaivev2, var_names=varnames);
az.summary(trace_tnaivev2, var_names=varnames)
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
             \anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
Got error No model on context stack. trying to find log_likelihood in translation.
Got error No model on context stack. trying to find log_likelihood in translation.
C:\Users\Mengj\anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
```

Out[18]:

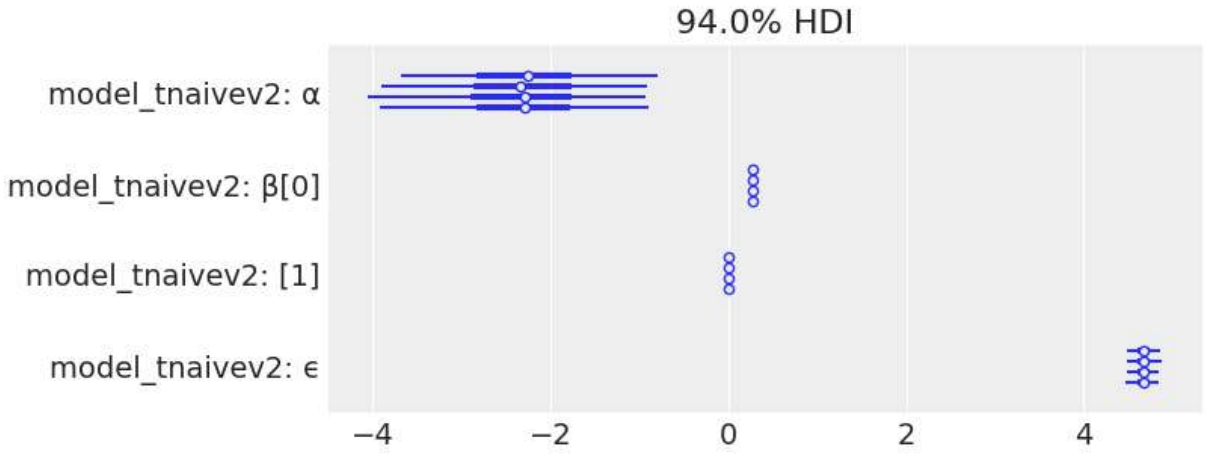
	mean	sd	hdi_3%	hdi_97%	mcse_mean	mcse_sd	ess_bulk	ess_tail	r_hat
α	-2.303	0.811	-3.863	-0.807	0.010	0.008	6527.0	2894.0	1.0
β[0]	0.267	0.010	0.249	0.287	0.000	0.000	6328.0	2917.0	1.0
β[1]	0.007	0.024	-0.037	0.052	0.000	0.000	7014.0	2837.0	1.0
ϵ	4.671	0.102	4.480	4.860	0.001	0.001	6251.0	2903.0	1.0



In [19]:
```
az.plot_forest([trace_tnaivev2],
               model_names=['model_tnaivev2'],
               var_names=['α', 'β', 'ϵ'],
               combined=False, colors='cycle', figsize=(8, 3))
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
              anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
```

Out[19]: array([<Axes: title={'center': '94.0% HDI'}>], dtype=object)



In [20]:
```
ppc_tv2 = pm.sample_posterior_predictive(trace_tnaivev2, samples=200, model=model_t
```

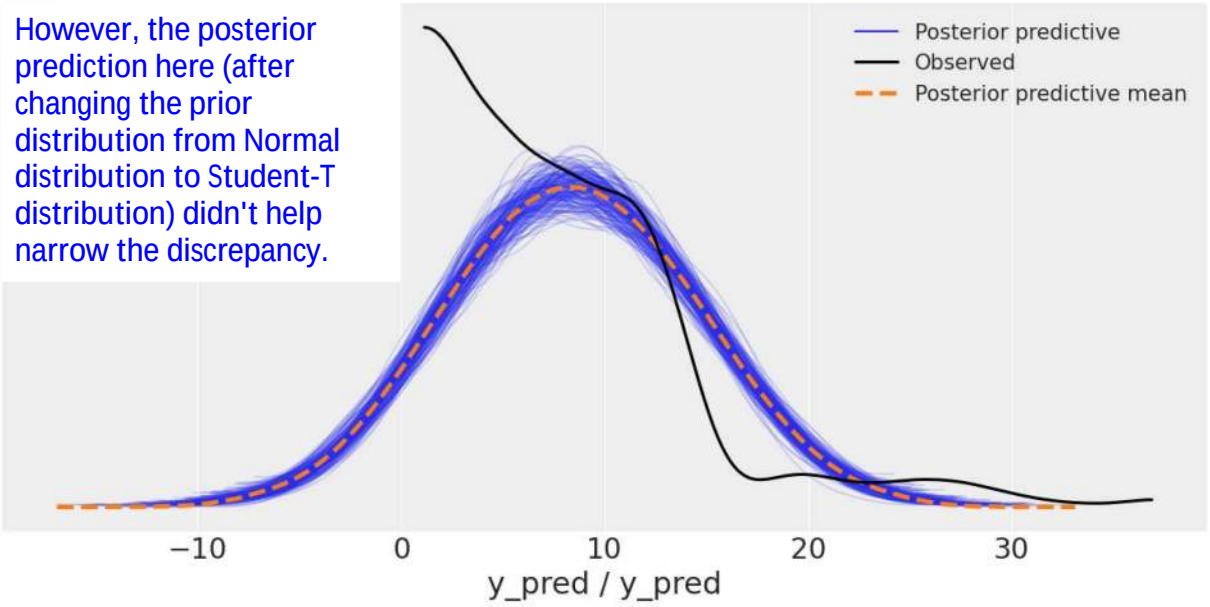
```
              anaconda3\envs\pm3env\lib\site-packages\pymc3\sampling.py:1708: UserW
arning: samples parameter is smaller than nchains times ndraws, some draws and/or ch
ains may not be represented in the returned posterior predictive sample
  warnings.warn(
```

100.00% [200/200 00:00<00:00]

In [21]:
```
data_ppc_tv2 = az.from_pymc3(trace=trace_tnaivev2, posterior_predictive=ppc_tv2)
ax = az.plot_ppc(data_ppc_tv2, figsize=(12, 6), mean=True)
# plt.xlim(-15, 35)
```

```
Got error No model on context stack. trying to find log_likelihood in translation.
              anaconda3\envs\pm3env\lib\site-packages\arviz\data\io_pymc3_3x.py:98:
FutureWarning: Using `from_pymc3` without the model will be deprecated in a future r
elease. Not using the model will return less accurate and less useful results. Make
sure you use the model argument or call from_pymc3 within a model context.
  warnings.warn(
posterior predictive variable y_pred's shape not compatible with number of chains an
d draws. This can mean that some draws or even whole chains are not represented.
```

However, the posterior prediction here (after changing the prior distribution from Normal distribution to Student-T distribution) didn't help narrow the discrepancy.



I tried some methods to see if there is a way to force y_pred to only positive numbers. I thought the exponential function might be helpful, but it failed...

Can I ask for some examples forcing the y_pred to positive numbers so that I can learn from it?

In [14]:
```
out_array_exp = np.exp(out_array)
out_array_exp
```

Out[14]:
```
array([5.61562168e+00, 8.55808891e+01, 3.52967138e+09, ...,
       9.07915203e+00, 5.10302499e+00, 7.44799598e+00])
```

In [15]:
```
with pm.Model() as model_exnaive:
    α_tmp = pm.Normal('α_tmp', mu=0, sd=10)
    β = pm.Normal('β', mu=0, sd=1, shape=2)
    ϵ = pm.HalfCauchy('ϵ', 5)

    μ = α_tmp + pm.math.dot(in_array, β)

    α = pm.Deterministic('α', α_tmp - pm.math.dot(x_mean, β))

    y_pred_temp = pm.Deterministic('y_pred_temp', pm.math.exp(μ))

    # y_pred = pm.Exponential('y_pred', 1/μ, observed=out_array)

    y_pred = pm.Normal('y_pred', mu=y_pred_temp, sd=ϵ, observed=out_array_exp)

    trace_exnaive = pm.sample(1000)
```

```
  anaconda3\envs\pm3env\lib\site-packages\deprecat\classic.py:215: Futu
reWarning: In v4.0, pm.sample will return an `arviz.InferenceData` object instead of
a `MultiTrace` by default. You can pass return_inferencedata=True or return_inferenc
edata=False to be safe and silence this warning.
  return wrapped_(*args_, **kwargs_)
Auto-assigning NUTS sampler...
Initializing NUTS using jitter+adapt_diag...
Multiprocess sampling (4 chains in 4 jobs)
NUTS: [ϵ, β, α_tmp]
```

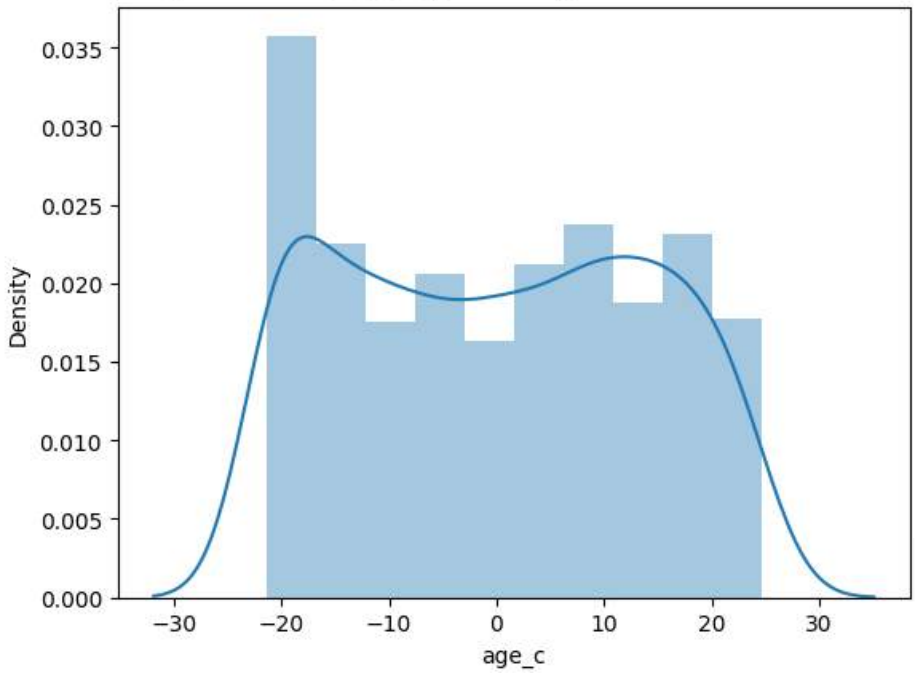
3.81% [305/8000 00:00<00:10 Sampling 4 chains, 0 divergences]

```
---------------------------------------------------------------------------
RemoteTraceback                           Traceback (most recent call last)
RemoteTraceback:
"""
Traceback (most recent call last):
  File                \anaconda3\envs\pm3env\lib\site-packages\pymc3\parallel_sampli
ng.py", line 137, in run
    self._start_loop()
  File                \anaconda3\envs\pm3env\lib\site-packages\pymc3\parallel_sampli
ng.py", line 191, in _start_loop
    point, stats = self._compute_point()
  File                \anaconda3\envs\pm3env\lib\site-packages\pymc3\parallel_sampli
ng.py", line 216, in _compute_point
    point, stats = self._step_method.step(self._point)
  File                \anaconda3\envs\pm3env\lib\site-packages\pymc3\step_methods\ar
raystep.py", line 276, in step
    apoint, stats = self.astep(array)
  File                \anaconda3\envs\pm3env\lib\site-packages\pymc3\step_methods\hm
c\base_hmc.py", line 147, in astep
    self.potential.raise_ok(self._logp_dlogp_func._ordering.vmap)
  File                \anaconda3\envs\pm3env\lib\site-packages\pymc3\step_methods\hm
c\quadpotential.py", line 268, in raise_ok
    raise ValueError("\n".join(errmsg))
ValueError: Mass matrix contains zeros on the diagonal.
The derivative of RV `β`.ravel()[0] is zero.
The derivative of RV `β`.ravel()[1] is zero.
The derivative of RV `α_tmp`.ravel()[0] is zero.
"""

The above exception was the direct cause of the following exception:

ValueError                                Traceback (most recent call last)
ValueError: Mass matrix contains zeros on the diagonal.
The derivative of RV `β`.ravel()[0] is zero.
The derivative of RV `β`.ravel()[1] is zero.
The derivative of RV `α_tmp`.ravel()[0] is zero.

The above exception was the direct cause of the following exception:

RuntimeError                              Traceback (most recent call last)
Cell In[15], line 16
     12 # y_pred = pm.Exponential('y_pred', 1/μ, observed=out_array)
     14 y_pred = pm.Normal('y_pred', mu=y_pred_temp, sd=ϵ, observed=out_array_exp)
---> 16 trace_exnaive = pm.sample(1000)

File ~\anaconda3\envs\pm3env\lib\site-packages\deprecat\classic.py:215, in deprecat.
<locals>.wrapper_function(wrapped_, instance_, args_, kwargs_)
    213         else:
    214             warnings.warn(message, category=category, stacklevel=_routine_st
acklevel)
--> 215 return wrapped_(*args_, **kwargs_)

File ~\anaconda3\envs\pm3env\lib\site-packages\pymc3\sampling.py:575, in sample(draw
s, step, init, n_init, initvals, trace, chain_idx, chains, cores, tune, progressbar,
model, random_seed, discard_tuned_samples, compute_convergence_checks, callback, jit
ter_max_retries, start, return_inferencedata, idata_kwargs, mp_ctx, pickle_backend,
```

```
**kwargs)
    573 _print_step_hierarchy(step)
    574 try:
--> 575     trace = _mp_sample(**sample_args, **parallel_args)
    576 except pickle.PickleError:
    577     _log.warning("Could not pickle model, sampling singlethreaded.")

File ~\anaconda3\envs\pm3env\lib\site-packages\pymc3\sampling.py:1496, in _mp_sample
(draws, tune, step, chains, cores, chain, random_seed, start, progressbar, trace, mo
del, callback, discard_tuned_samples, mp_ctx, pickle_backend, **kwargs)
   1494 try:
   1495     with sampler:
-> 1496         for draw in sampler:
   1497             trace = traces[draw.chain - chain]
   1498             if trace.supports_sampler_stats and draw.stats is not None:

File ~\anaconda3\envs\pm3env\lib\site-packages\pymc3\parallel_sampling.py:479, in Pa
rallelSampler.__iter__(self)
    476     self._progress.update(self._total_draws)
    478 while self._active:
--> 479     draw = ProcessAdapter.recv_draw(self._active)
    480     proc, is_last, draw, tuning, stats, warns = draw
    481     self._total_draws += 1

File ~\anaconda3\envs\pm3env\lib\site-packages\pymc3\parallel_sampling.py:359, in Pr
ocessAdapter.recv_draw(processes, timeout)
    357     else:
    358         error = RuntimeError("Chain %s failed." % proc.chain)
--> 359     raise error from old_error
    360 elif msg[0] == "writing_done":
    361     proc._readable = True

RuntimeError: Chain 0 failed.
```

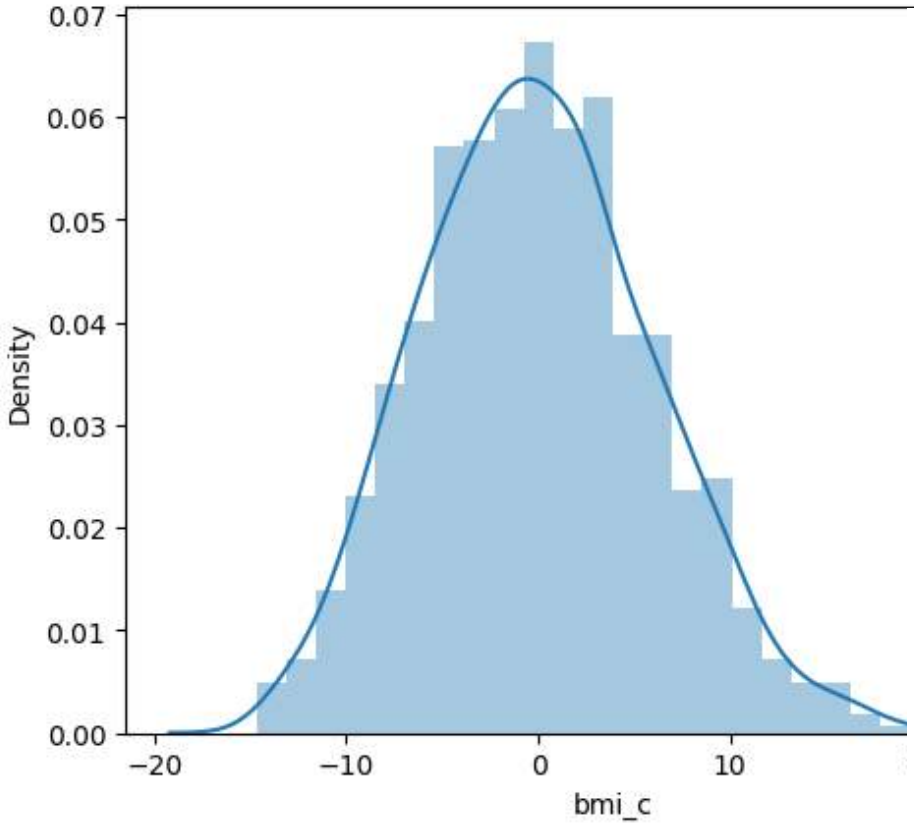
In I also plotted the distribution of the variables, next I want to try (and I am wondering if the following idea reasonable at all) --

```
sns.distplot(nsmokerframe['age_c'], kde=True);
```



I want to have some assumption on my independent variables based on my observation of the distribution here (assuming that my observation is my knowledge of the data), then sample the independent variables.

The data contains more people in young age group, but I want to try put the uniform distribution here, range around [-22, 25].

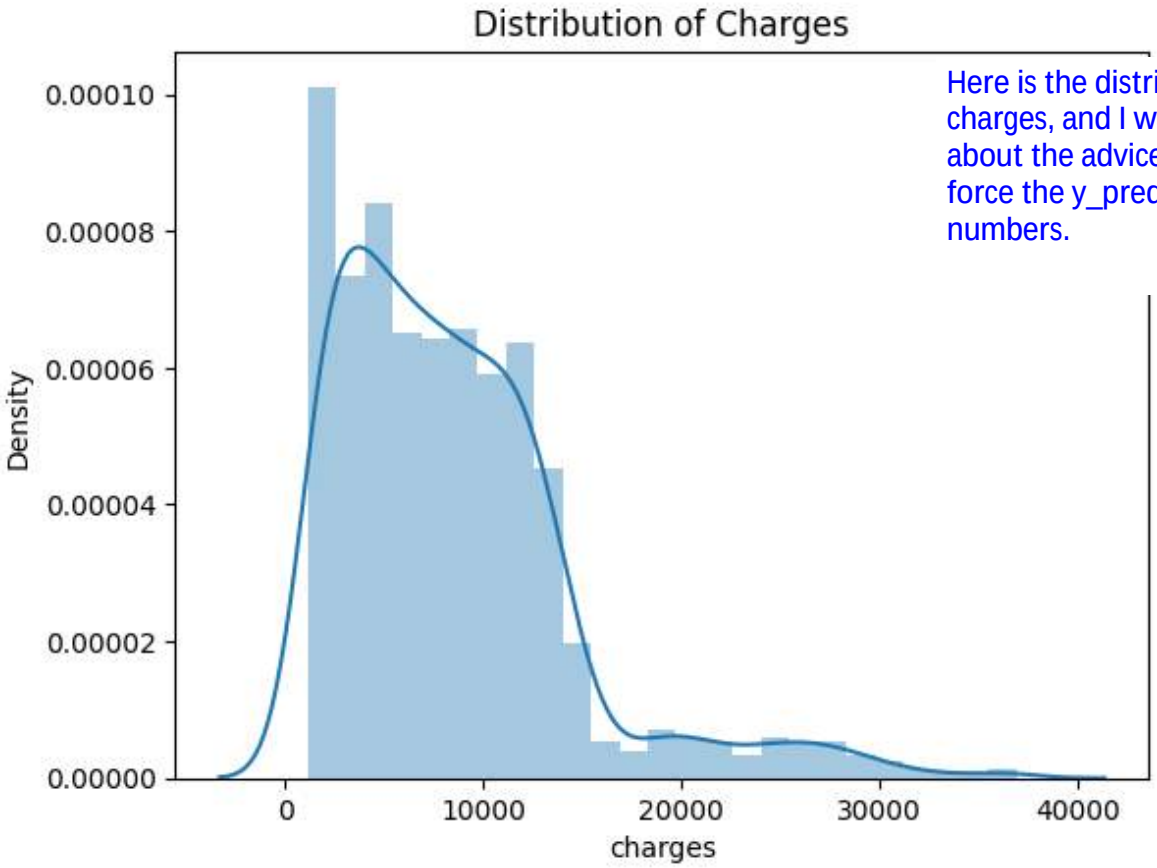


The for the data of centered bmi, I will assume the normal distribution.

Then I want to get the posterior predictive samples of these two variables to see if the predictions will agree with the data.

If it does, I want to use my samples as the input data to predict the charges.

I am actually not sure why have to do this... I am just curious if my target prediction (the insurance charges in this dataset) can be improved by doing this.



Here is the distribution of the charges, and I want to ask about the advice on how to force the y_pred to positive numbers.