```
import matplotlib.pyplot as plt
import matplotlib.gridspec as gs

def genHypos(min, max, P3=False):
    hypoDict = dict()

    hypoDict['H1'] = list(range(min + 1, max + 1, 2)) #even numbers
    hypoDict['H2'] = list(range(min, max + 1, 2)) #odd numbers
    hypoDict['H3'] = [x**2 for x in range(min, max + 1) if x**2 <= 100] #square numbers
    hypoDict['H4'] = [x for x in range(min + 1, max + 1) if all(x % i != 0 for i in range(min +
1, x))] #prime numbers
    hypoDict['H5'] = [x * 5 for x in range(min, max + 1) if x*5 <= 100] #multiples of 5
    hypoDict['H6'] = [x * 10 for x in range(min, max + 1) if x*10 <= 100] #multiples of 10
    hypoDict['H7'] = list(range(min, max + 1)) #all numbers

    #below code generates hypotheses H8 to H4957 for problem 3:
    if P3:
        start = 8
        limit = 100
        for i in range(1, limit + 1):
            for j in range(0, limit):
                name = 'H' + str(start)
                lst = list(range(i, limit + 1 - j))
                if len(lst) > 1:
                    hypoDict[name] = lst
                    start += 1

    return hypoDict

HYPOTHESIS_DICT = genHypos(1, 100)
DATA_SETS = [[], [50], [53], [50, 53], [16], [10, 20], [2, 4, 8], [2, 4, 8, 10]]

"""Problem 1:
    Write a function that takes an argument x and a hypothesis (however you represent
it)
    and computes a size principle likelihood (e.g. where the likelihood of each number in
    the set is equal). Write down what likelihood each hypothesis assigns to each data
    point in it. What does each hypothesis assign to data points not in it?

    ANSWER:
        Likelihoods for data points in each hypothesis:
            P(D ∈ H1 | H1) = 0.02 = 1/50
            P(D ∈ H2 | H2) = 0.02 = 1/50
            P(D ∈ H3 | H3) = 0.1 = 1/10
            P(D ∈ H4 | H4) = 0.04 = 1/25
            P(D ∈ H5 | H5) = 0.05 = 1/20
```

```
        P(D ∈ H6 | H6) = 0.1 = 1/10
        P(D ∈ H7 | H7) = 0.01 = 1/100
        P(D ∉ Hn | Hn) = 0 (for data points not in the hypothesis)
    """

def likelihood(dataSet, hypothesis):
    """Returns the P(dataSet | Hx) where Hx is a hypothesis"""
    if not dataSet:
        return 1.0

    hypo = HYPOTHESIS_DICT[hypothesis]

    for d in dataSet:
        if d not in hypo:
            return 0

    return 1 / len(hypo) ** len(dataSet)

"""Problem 2:
    Make a plot showing the posterior predictive probability (marginalizing over
hypotheses)
    that each number 1...100 is “in” the concept, for each of the following data sets:

    (a) No data
    (b) 50
    (c) 53
    (d) 50, 53
    (e) 16
    (f) 10, 20
    (g) 2, 4, 8
    (h) 2, 4, 8, 10

    Assume that there are equal priors on each hypothesis and the size principle
likelihood
    from Q1. Be sure that you structure your code to process the data (a)-(h) as a list and
    do not “hard code” in these datasets (i.e. the likelihood function and posterior function
    should accept any list of data). Write a sentence for each plot about whether the
model
    does or does not capture your intuitions about the “right” answer.

    ANSWER:
        For all plots, my intuition is that the probabilities for each number in
        the given range will have a probability that is proportional to its similarity
        to the given data given the set of hypotheses. I will leave out the H7 hypothesis
        in most answers (except for [50,53] because all numbers fit into this hypothesis.

        (Plot a) No data:
```

With no data to test likelihood with, each hypothesis has equal probability (just the prior). The plot shows this as each number in the prediction range has the prior probability (1/7 in this case) summed over the number of hypotheses the number falls into. For example, 100 has the highest probability of ~.71 because it falls into 5 hypotheses and 5*1/7 is ~.71. So the plot does capture my intuition about the correct answer.

(Plot b) 50:
This plot shows 4 different probability bars because 50 is in 4 hypotheses. Multiples of 10 all have P=1 because they are all captured by the same hypotheses as 50. Multiples of 5 have the next highest P because they fit into 3 of the 4 possible hypotheses.

(Plot c) 53:
This plot is similar to plot b but with different possible hypotheses. The most likely hypothesis is H4, prime numbers. Numbers that are odd and prime have P=1.

(Plot d) 50, 53:
In this plot, all numbers have equal probability of 1. Because 50 and 53 only have
H7 in common, we can infer that all numbers are in the unknown data.

(Plot e) 16:
This plot shows the P=1 for square and even numbers with odd squares being next most likely. This makes intuitive sense because even squares have the most
in common with 16 so they are equally as likely to be true.

(Plot f) 10, 20:
10 and 20 are both even mulitiples of 5 and 10 so it makes intuitive sense that 30, 40,... 100 would also have P=1 and the plot shows this. Next most probable
are even multiples of 5.

(Plot g) 2, 4, 8:
see plot h explanation.

(Plot h) 2, 4, 8, 10:
I think plots g and h are the most informative for how inferencing works. The two
data sets include the same numbers but h has an additional number (10) that fits into
the same hypothesis as set g. The plots are nearly identical but in plot h, H7 'all numbers'
is about half as likely as it was in plot g. The one extra data point has a big effect on the probability distribution even though it didn't rule out any

hypotheses
that were seen in plot g, it made H1 a much more probable hypothesis than H7.
"""

```
def bayesRule2(dataList):
    """Returns a dictionary of hypotheses as keys and P(h|dataList) as values """
    prior_prob = 1 / len(HYPOTHESIS_DICT) #P(h)
    hypo_given_data = dict()
    norm = 0

    for key in HYPOTHESIS_DICT:
        hypo_given_data[key] = likelihood(dataList, key) * prior_prob
        norm += hypo_given_data[key]

    for key in hypo_given_data:
        hypo_given_data[key] /= norm

    return hypo_given_data


def pos_pred_prob(dataList, func, start, finish):

    hypo_given_data = func(dataList)
    xList = list(range(start, finish + 1))
    prob_list = []

    for x in xList:
        total = 0
        for key in HYPOTHESIS_DICT:
            if x in HYPOTHESIS_DICT[key]:
                total += hypo_given_data[key] * 1

        prob_list += [total]

    return xList, prob_list

def plotProbs(dataList, title, func=bayesRule2):

    fig = plt.figure(figsize=(20, 10))
    grid = gs.GridSpec(3, 3, figure=fig)
    i, j, k = 0, 0, 0
    axList = []
    colorList = ['red', 'blue', 'green', 'white', 'cyan', 'orange', 'magenta', 'pink']

    for d in dataList:
        x, y = pos_pred_prob(d, func, 1, 100)
```

```
        axs = fig.add_subplot(grid[i, j])
        axs.bar(x, y, color=colorList[k])
        axs.set_facecolor('black')
        k += 1

        axs.set_title('Posterior Probability for ' + str(d), fontsize=10, y=.98)

        if j == 0:
            axs.set_ylabel('Probability', fontsize=12)

        if (i == 1 and j== 1) or i == 2:
            axs.set_xlabel('Range', fontsize=12)

        j += 1
        if j == 3:
            i += 1
            j = 0

        if i == 2 and j == 1:
            j = 2
    fig.suptitle(title, fontsize=15, y=.95)
    # plt.savefig('Prob3_plot_FULL.pdf')
    plt.show()

title = 'P2: Posterior Predictive Probabilities, Marginalized Over All Hypotheses'
plotProbs(DATA_SETS, title)


"""Problem 3:
    Re-make the plots from Q2 but now incorporate range-based hypotheses. To do this,
assume
    that H1-H7 each have a prior of 1/8, and the remaining 1/8th probability is distributed
    equally among all intervals in the range 1-100. Here we will define as “interval” as
something
    containing two distinct points (e.g. [50-51] or [3-88] but not [31]); there are a lot of
    these range-based hypotheses-- how many are there? Write a sentence for each plot
about
    whether the including the range-based hypotheses makes them better match your
own intuitions
    about how to generalize and why.

    ANSWER:
        There are 4950 range based hypotheses. For all plots, including range based
hypotheses
        made them fit my intuition about how humans generalize: linearly on the number
line.
        These plots illustrate 'anchoring bias' or 'anchoring heuristic', an effect seen in
```

behavioral economics whereby people use initial data (like the numbers in our data sets)
as a reference point which then influences decisions made later regarding the data.
As an example, in negotiating pay for a job or contract, if the employer offers $55k the employee will counter amounts around $55k. They will likely not counter with $100k
even if the employer's offer is unreasonably low.

All of these plots have similar shape to those in problem 2 but the probabilities are skewed towards the given data; we generalize linearly based on the number line. This is
particularly interesting because the probabilities for each additional hypothesis is only
(1/8)/4950 and yet they have a big effect on the probabilities.

(Plot a) No data:
Here we see the probabilities skew towards the shape of a normal distribution.
With no data, each number's probability is as likely as the number of hypotheses
it falls into times the prior. The mean of the range, 50, will fall into more
of the range based hypotheses so the probabilities for numbers on either side is
lower than in problem 2's plot.

(Plot b) 50:
50 is now most likely (P=1) and probability of numbers on either side are lower than in
problem 2. Given the number 50, my intuition would be to guess that other unknown
numbers are closer to 50 on the number line.

(Plot c) 53:
The same effect is seen as in plot b but the maximum probability is now at 53 for
the same reasons.

(Plot d) 50, 53:
This is probably the most interesting plot in problem 3. The interval around 50-53
now has P=1 and all other numbers have dropped in probability compared to problem 2.
Again, this shows the anchoring effect and generalizing based on the number line.

(Plot e) 16:

The plot is centered around 16 for the same reason as in other plots.

(Plot f) 10, 20:
The skewing towards the interval 10-20 is more difficult to see in this plot because it is large but 10 and 20 have P=1 and other multiples of 10 are slightly
lower because they are not in the 10-20 interval.

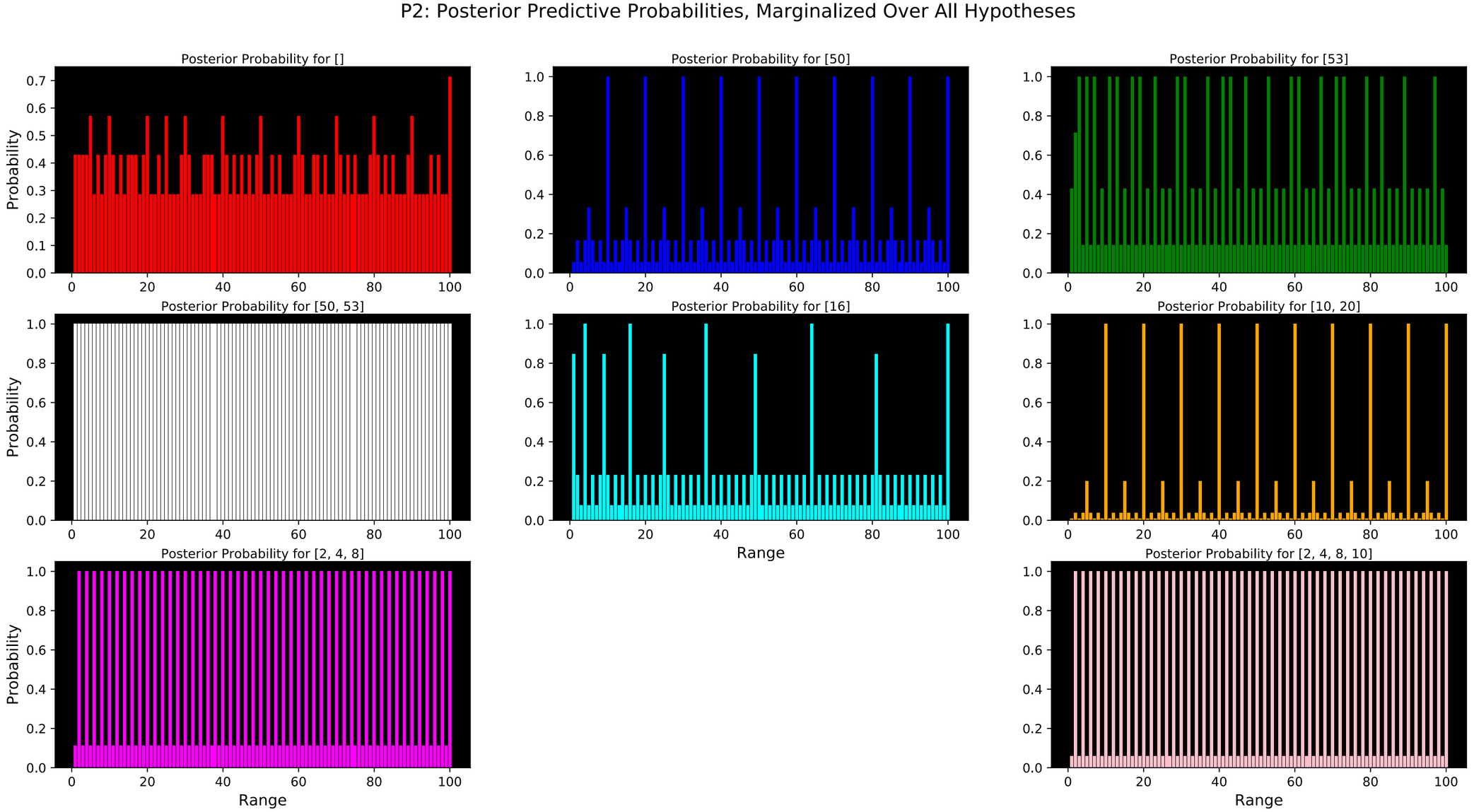
(Plot g) 2, 4, 8:
Please see explanation for plot h.

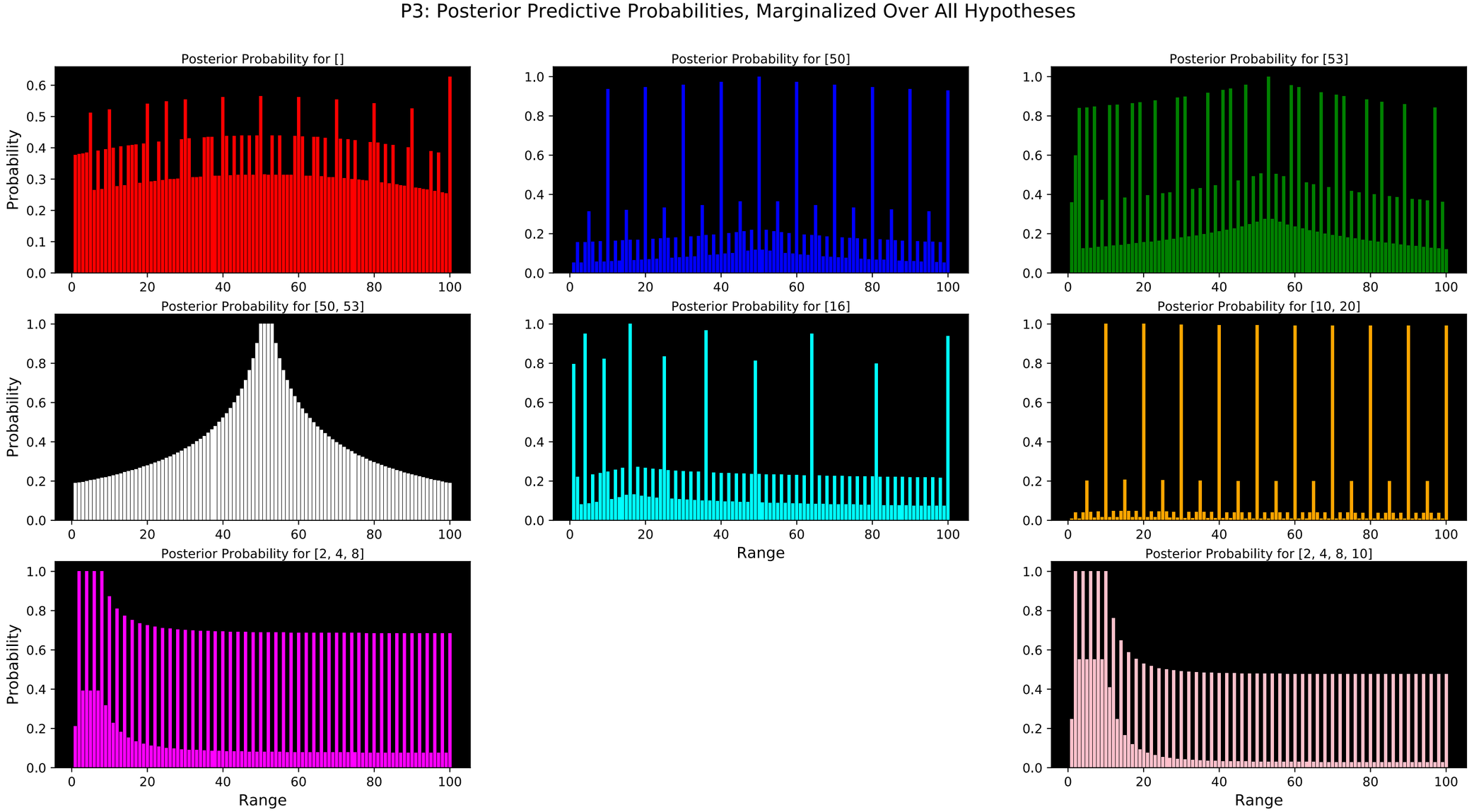
(Plot h) 2, 4, 8, 10:
Non even numbers in the interval 2-10 are now more likely than in problem 2 and even numbers outside the interval now have lower probability than they did in
problem 2. There is more weight on that interval than on other hypotheses which again
displays how humans would generalize on to the number line rather than other possible
hypotheses.

"""

```
def bayesRule3(dataList):
    """Returns a dictionary of hypotheses as keys and P(h|dataList) as values """
    prior_prob = 1 / 8 #P(h)
    next_prob = (1 / 8) / (len(HYPOTHESIS_DICT) - 7) #all 'H8' hypos share 1/8 prob equally
    hypo_given_data = dict()
    norm = 0
    add_hypos = ['H' + str(i) for i in range(8, 4958)]

    for key in HYPOTHESIS_DICT:
        if key in add_hypos:
            hypo_given_data[key] = likelihood(dataList, key) * next_prob
            norm += hypo_given_data[key]
        else:
            hypo_given_data[key] = likelihood(dataList, key) * prior_prob
            norm += hypo_given_data[key]

    for key in hypo_given_data:
        hypo_given_data[key] /= norm

    return hypo_given_data

HYPOTHESIS_DICT = genHypos(1, 100, P3=True)
```

```
title = 'P3: Posterior Predictive Probabilities, Marginalized Over All Hypotheses'
plotProbs(DATA_SETS, title, bayesRule3)
```


P2: Posterior Predictive Probabilities, Marginalized Over All Hypotheses
Posterior Probability for []
Posterior Probability for [50]
Posterior Probability for [53]
Posterior Probability for [50, 53]
Posterior Probability for [16]
Posterior Probability for [10, 20]
Posterior Probability for [2, 4, 8]
Posterior Probability for [2, 4, 8, 10]
Probability
Range


P3: Posterior Predictive Probabilities, Marginalized Over All Hypotheses
Posterior Probability for []
Posterior Probability for [50]
Posterior Probability for [53]
Posterior Probability for [50, 53]
Posterior Probability for [16]
Posterior Probability for [10, 20]
Posterior Probability for [2, 4, 8]
Posterior Probability for [2, 4, 8, 10]
Probability
Range