УДК 331.08

**Выбор метода для определения параметров модели Басса прогнозирования развития возобновляемой энергетики на примере ветрогенерации на языке программирования Python с использованием библиотеки scipy**

Никифоров М.М., Куделин А.Г. (nikiforov1601@gmail.com)

*Ухтинский государственный технический университет, г. Ухта, Россия*

1. Аннотация
2. Описание проблемы
3. Выбор языка программирования и библиотек
4. Описание подхода к прогназированию
5. Описание применяемых методов
6. Выбор наилучшего метода
7. Заключение

При выборе модели моделирования и прогнозирования на основе модели Басса, существует не мало методов для осуществления её работы. Каждый из методов является оптимальным в зависимости от данных, с которыми работает модель. Необходимо исследовать методы, которые можно применить к выбранной модели и выбрать тот метод, который даёт самое оптимальное решение.

В качестве модели Басса, было использовано следующее уравнение [4]:

Где S(t) – продажи в период t; – совокупные продажи за период [0 ... t - 1]; p - коэффициент инновации, q - коэффициент имитации, m - общее количество всех покупок.

Модель Басса описывает продажи за период, которые являются производной от ветрогенерации за прошлый период. Определив продажи за период, мы высчитываем текущую ветрогенерацию. Таким образом, определив параметры p, q, m, так, что смоделированные данные будут максимально равны фактическим, мы в дальнейшем можем давать прогноз на необходимые промежутки времени.

Для реализации моделирования и прогнозирования был выбран язык программирования – python, являющийся одним из основных используемых в научных вычислениях и имеющий множество библиотек, специализированных для работы с математическими вычислениями. Для решения поставленной задачи, были выбраны следующие библиотеки:

- numpy (Numerical Python), для работы с массивами, матрицами;

- pandas, для работы с DataFrame;

- scipy, предназначенная для выполнения научных и инженерных расчётов.

Функция minimize (минимизация скалярной функции одной или нескольких переменных) модуля scipy.optimize отлично подойдет для оптимального подбора параметров p, q, m, так, что полученные данные будут максимально приближены к фактическим. Данная функция может минимизировать параметры функции разными методами:

* Nelder-Mead
* Powell
* CG
* BFGS
* Newton-CG
* L-BFGS-B
* TNC
* COBYLA
* SLSQP
* trust-constr
* dogleg
* trust-ncg
* trust-exact
* trust-krylov

Рассмотрим каждый из методов в отдельности.

Nelder-Mead – также известный как метод деформируемого многогранника и симплекс-метод, — метод безусловной оптимизации функции от нескольких переменных, не использующий производной (точнее — градиентов) функции, а поэтому легко применим к негладким и/или зашумлённым функциям.

Powell – представляет собой алгоритм, предложенный Майклом Дж. Д. Пауэллом для нахождения локального минимума функции. Функция не обязательно должна быть дифференцируемой.

CG – Минимизация скалярной функции одной или нескольких переменных с использованием алгоритма сопряженных градиентов

BFGS – итерационный метод численной оптимизации, предназначенный для нахождения локального максимума/минимума нелинейного функционала без ограничений. Один из наиболее широко применяемых квазиньютоновских методов.

Newton-CG – Метод сопряжённых градиентов (Метод Флетчера — Ривcа) — метод нахождения локального экстремума функции на основе информации о её значениях и её градиенте.

L-BFGS-B – модификация метода BFGS с ограниченным использованием памяти в многомерном кубе

TNC – Минимизируйте скалярную функцию одной или нескольких переменных, используя усеченный алгоритм Ньютона

COBYLA – Минимизируйте скалярную функцию одной или нескольких переменных, используя алгоритм Ограниченной оптимизации с помощью линейной аппроксимации

SLSQP – Минимизируйте скалярную функцию одной или нескольких переменных с помощью последовательного программирования методом наименьших квадратов

trust-constr – Минимизация скалярной функции с учетом ограничений

dogleg – Минимизация скалярной функции одной или нескольких переменных с использованием алгоритма доверительной области изгиба

trust-ncg – Алгоритм доверительной области (trust region) сопряженных градиентов (Ньютона). Плохая обусловленность матрицы Гессе и неверные направления поиска могут привести к тому, что алгоритм сопряженных градиентов Ньютона может быть неэффективным. В таких случаях предпочтение отдается методу trust-ncg

trust-exact – Минимизация скалярной функции одной или нескольких переменных с использованием почти точного алгоритма доверительной области, который требует только произведения матричных векторов с матрицей Гессе

trust-krylov – Минимизация скалярной функции одной или нескольких переменных с использованием почти точного алгоритма доверительной области, который требует только произведения матричных векторов с матрицей Гессе

Выделим среди указанных методов те, которые наиболее оптимально подходят для решения нашей задачи: Nelder-Mead, Powell, L-BFGS-B.

Исследуем каждый из методов на наборе данных по ветрогенерации в мировой генерации, Европе, Северной Америке, Центральной и южной Америке, Азиато-Тихоокеанском регионе, Среднем востоке.

По каждому исследованию сравним сумму квадратов разности фактических данных и спрогнозированных.

<https://habr.com/ru/post/439288/>

Нахождение частных проявлений этого общего закона, т. е. установление связей между различными явлениями, – одна из основных задач всякой науки…

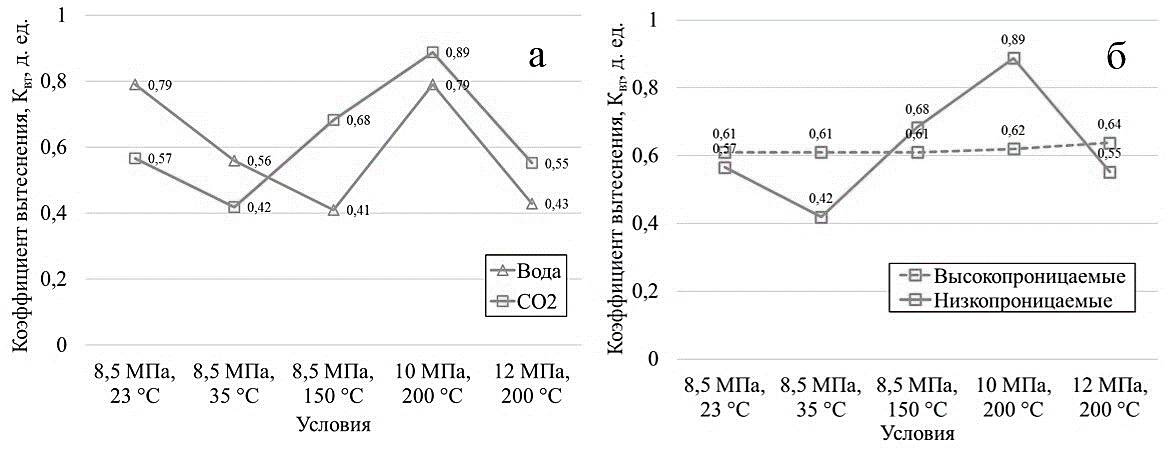
Для выполнения данной работы использовались следующие геофизические параметры (таблица 1):

Таблица 1 - Перечень геофизических параметров

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Параметр | Ед. изм. | Описание |
| 1 | ПЗ | омм | Кажущееся сопротивление, измеренное потенциал-зондом |
| 2 | ПС | мВ | Метод самопроизвольной поляризации |

\* - если отсутствует ННКб, то используется НГК и наоборот

В рамках исследований выполнено 15 экспериментов при различных термобарических условиях, соответствующих различным зонам пермо-карбоновой залежи Усинского месторождения (таблица 3). Результаты выполненных экспериментов приведены на рисунках 5 и 6.



*а*) влияние агента вытеснения, *б*) влияние газопроницаемости

Рисунок 1 – Сопоставление коэффициентов вытеснения

В рамках исследований выполнено ….

**Библиографический список:**

[4] F.M. Bass, Bass 1969 New Prod Growth Model.pdf, Manage. Sci. 15 (1969) 215–227.

1. Анализ эффективности применения долот на нефтегазовых скважинах в пределах непского свода / Р.У. Сираев, Р.Х. Акчурин, В.В. Че, А.Г. Вахромеев // ВЕСТНИК ИрГТУ. – 2013. - №5. – с.72-77. Текст: непосредственный.
2. Классификация PDC и алмазных долот и бурголовок по коду IADC [Электронный ресурс]. - <http://burintekh.ru/upload/iblock/783/783d2b431d89083ed5a23289ffff0e75.pdf> (дата обращения).

Не более 5 позиций.

‘Nelder-Mead’ [(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-neldermead.html#optimize-minimize-neldermead) - Минимизация скалярной функции одной или нескольких переменных с использованием алгоритма Нелдера-Мида

‘Powell’ [(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-powell.html#optimize-minimize-powell) - Минимизация скалярной функции одной или нескольких переменных с использованием модифицированного алгоритма Пауэлла

‘CG’ [(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-cg.html#optimize-minimize-cg) - Минимизация скалярной функции одной или нескольких переменных с использованием алгоритма сопряженных градиентов

‘BFGS’ [(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-bfgs.html#optimize-minimize-bfgs) - Минимизация скалярной функции одной или нескольких переменных с использованием алгоритма BFGS

‘Newton-CG’ [(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-newtoncg.html#optimize-minimize-newtoncg) - Минимизация скалярной функции одной или нескольких переменных с использованием алгоритма Newton-CG.

Обратите внимание, что параметр jac (Jacobian) обязателен.

‘L-BFGS-B’ [(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-lbfgsb.html#optimize-minimize-lbfgsb) - Минимизируйте скалярную функцию одной или нескольких переменных с помощью алгоритма L-BFGS-B.

‘TNC’ [(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-tnc.html#optimize-minimize-tnc) - Минимизируйте скалярную функцию одной или нескольких переменных, используя усеченный алгоритм Ньютона (TNC).

‘COBYLA’ [(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-cobyla.html#optimize-minimize-cobyla) - Минимизируйте скалярную функцию одной или нескольких переменных, используя алгоритм Ограниченной оптимизации с помощью линейной аппроксимации (COBYLA).

‘SLSQP’ [(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-slsqp.html#optimize-minimize-slsqp) - Минимизируйте скалярную функцию одной или нескольких переменных с помощью последовательного программирования методом наименьших квадратов (SLSQP).

‘trust-constr’[(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-trustconstr.html#optimize-minimize-trustconstr) - Минимизируйте скалярную функцию с учетом ограничений.

‘dogleg’ [(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-dogleg.html#optimize-minimize-dogleg) - Минимизация скалярной функции одной или нескольких переменных с использованием алгоритма доверительной области изгиба.

‘trust-ncg’ [(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-trustncg.html#optimize-minimize-trustncg) - Минимизация скалярной функции одной или нескольких переменных с использованием алгоритма Ньютона сопряженного градиента доверительной области.

‘trust-exact’ [(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-trustexact.html#optimize-minimize-trustexact) - Минимизация скалярной функции одной или нескольких переменных с использованием почти точного алгоритма доверительной области, который требует только произведения матричных векторов с матрицей Гессе.

‘trust-krylov’ [(see here)](https://docs.scipy.org/doc/scipy/reference/optimize.minimize-trustkrylov.html#optimize-minimize-trustkrylov) - Минимизация скалярной функции одной или нескольких переменных с использованием почти точного алгоритма доверительной области.