

UNIVERSIDADE FEDERAL DO PARANÁ

CAROLINE RUSCH SCHULZE GRR20180876

EMANUELY DA SILVA SANTOS GRR20180818

NUTRIÇÃO DE BOVINOS LEITEIROS EM PYTHON

CURITIBA

2019

CAROLINE RUSCH SCHULZE GRR20180876
EMANUELY DA SILVA SANTOS GRR20180818

NUTRIÇÃO DE BOVINOS LEITEIROS EM PYTHON

Trabalho apresentado ao curso de Programação de computadores, Setor de informática, Universidade Federal do Paraná, como requisito parcial à obtenção do título de Mestre em Informática.

Orientador(a): Prof MSc Jackson Antonio do Prado Lima.

CURITIBA
2019

SUMÁRIO:

1-Resumo

2-Introdução

3-Fundamentação Teórica

4-Listas de figuras, tabelas utilizadas no trabalho

5-Resultados obtidos com a implementação

6-Dificuldades encontradas durante o trabalho

7-Sugestões de trabalhos para serem aplicados na disciplina.

1-Resumo:

O trabalho foi desenvolvido em código de Python com a finalidade de ajudar o homem no campo ou que trabalha na área de bovino leiteiro, disponibilizando um Bot no Telegram que fornece as informações nutricionais de acordo com o peso e período que a vaca se encontra, facilitando o manejo do animal e alimentando-o corretamente.

O trabalho foi feito em conjunto de duas alunas de agronomia durante a matéria de Zootecnia geral, que explica bastante a importância do manejo nutricional adequado ligado com a boa qualidade de vida e maior eficiência produtiva do gado leiteiro.

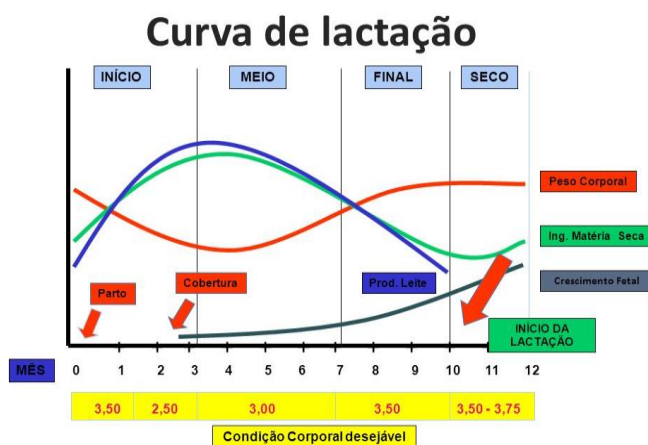
2- Introdução:

Com objetivo de ter funcionalidade no mercado e aplicar os conhecimentos em Python, a principal motivação foi ver a possibilidade de trabalhar em conjunto com o público alvo de agronomia (os produtores rurais), mesmo quando não for possível a presença de um engenheiro agrônomo para consultoria.

A problemática se concentra na falta de informação de agricultores e pecuaristas para ter as exatas quantidades de cada nutriente que o animal ou planta necessitam para realizar suas funções corretamente.

Se for a primeira vez que a vaca for emprenhar, a lactação não se iniciará simultaneamente com a fecundação, porém a necessidade nutricional é a considerada a mesma.

período seco: não deve retirar leite da vaca.



4-Listas de figuras, tabelas utilizadas no trabalho:

Na figuras 1,2 e 3 é foi feita uma tabela no Excel com cada valor a cada período diante do peso aproximado fornecido

1	LEGENDA:Exigências em energia metabolizável (EM), nutrientes					
2	digestíveis totais (NDT), proteína bruta (PB), cálcio (Ca) e fósforo					
3	(P)					
4						
5	Sua vaca possui 10 meses ou mais? (NÃO= "Seu animal ainda não					
6	alcançou a puberdade" SIM=(Se possui peso vivo entre 250 e					
7	280 kg=continua o programa se não="Sua vaca alcançou a					
8	puberdade, mas zootecnicamente não está preparada para a					
9	monta"))					
10						
11						
12						
13	Manutenção de vacas no início da lactação					
14	Peso vivo	Em(Mcal)	NDT(kg)	PB	Ca(kg)	P(kg)
15	400	11,39	2,96	0,299	0,015	0,011
16	450	12,87	3,31	0,329	0,017	0,012
17	500	13,74	3,46	0,338	0,02	0,014
18	550	14,13	3,83	0,354	0,023	0,015
19	600	15,61	4,15	0,376	0,025	0,015
20	650	16,2	4,34	0,402	0,026	0,016
21	700	17,58	4,56	0,424	0,027	0,018
22						100

Comentado [CRS1]:

Figura 1

	A	B	C	D	E	F	G
24							
25	Manutenção de vacas no meio da lactação						
26	Peso vivo	Em(Mcal)	NDT(kg)	PB	Ca(kg)	P(kg)	Uréia(g/dia)
27	400	12,01	3,13	0,318	0,016	0,011	
28	450	13,12	3,42	0,341	0,018	0,013	
29	500	14,2	3,7	0,364	0,02	0,014	
30	550	15,25	3,97	0,386	0,022	0,016	
31	600	16,28	4,24	0,406	0,024	0,017	
32	650	17,29	4,51	0,428	0,026	0,019	
33	700	18,28	4,76	0,449	0,028	0,02	120
34							
35	Manutenção de vacas em final de gestação						
36	(kg)	Em (Mcal)	NDT (kg)	(kg)	Ca (kg)	P (kg)	Uréia(g/dia)
37	400	15,26	4,15	0,89	0,026	0,016	
38	450	16,66	4,53	0,973	0,03	0,018	
39	500	18,04	4,9	1,053	0,033	0,02	
40	550	19,37	5,27	1,131	0,036	0,022	
41	600	20,68	5,62	1,207	0,039	0,024	
42	650	21,96	5,97	1,281	0,043	0,026	
43	700	23,21	6,31	1,355	0,046	0,028	120
44							

Figura 2

44							
45	Manutenção de vacas secas						
46	(kg)	Em (Mcal)	NDT (kg)	(kg)	Ca (kg)	P (kg)	Uréia(g/dia)
47	400	16,14	4,37	0,97	0,031	0,019	
48	450	17,74	4,74	1,007	0,036	0,021	
49	500	18,49	4,99	1,197	0,042	0,025	
50	550	19,03	5,39	1,2	0,049	0,028	
51	600	19,71	5,82	1,264	0,053	0,032	
52	650	20,42	6,17	1,299	0,057	36	
53	700	22,23	6,52	1,384	0,062	0,038	100
54							
55							
56							

Figura 3

Transferindo isso para uma matriz em python (Figura 4), vendo que foi criada uma variável contendo uma lista do peso(como se fosse o eixo das ordenadas na tabela), outra variável para cada nutriente(eixo das abscissas) e como se criasse uma tabela para cada período que a vaca se encontra.

```

1 from datetime import date
2 pesos = [400, 450, 500, 550, 600, 650, 700]
3 exigencias=["Em", "MDT", "PB", "Ca", "P"]
4 inicio_lac=[11.39, 2.96, 0.299, 0.015, 0.011], [12.87, 3.31, 0.329, 0.017, 0.012], [13.74, 3.46, 0.338, 0.02, 0.014], [14.13, 3.83, 0.354, 0.023, 0.015], [15.
5 meio_lac=[12.01, 3.13, 0.318, 0.016, 0.011], [13.12, 3.42, 0.341, 0.018, 0.013], [14.2, 3.7, 0.364, 0.02, 0.014], [15.25, 3.97, 0.386, 0.022, 0.016], [16.28, 4
6 final_gest=[15.26, 4.15, 0.89, 0.026, 0.016], [16.66, 4.53, 0.973, 0.03, 0.018], [18.04, 4.9, 1.053, 0.033, 0.02], [19.37, 5.27, 1.131, 0.036, 0.022], [20.66
7 vacas_secas=[16.14, 4.37, 0.97, 0.031, 0.019], [17.74, 4.74, 1.007, 0.036, 0.021], [18.49, 4.99, 1.197, 0.042, 0.025], [19.03, 5.39, 1.2, 0.049, 0.028], [19.7

```

Figura 4

Se a vaca não estiver apta para receber monta, receberá uma mensagem contendo essa informação e esperar a data para ser eficaz.

No início do programa, foi importado uma biblioteca que reconhece o dia presente, para depois ser usado para ver quantos dias de gestação a vaca está. A função leva como parâmetro o t, que é o dia, e a multiplicação com o 360 é para deixar em dias, considerando que multiplicou por dias do mês (30). Figuras 5 e 6.

```

1 from datetime import date

```

Figura 5

```

27 def dias_gestação(t):
28     hoje=date.today()
29     tempo_dias_monta = int(t[0]) + (int(t[1]) * 30) + (int(t[2]) * 365)
30     tempo_dias_hoje = (int(hoje.day)) + ((int(hoje.month)) * 30) + ((int(hoje.year)) * 365)
31     tempo_prenhez = tempo_dias_hoje - tempo_dias_monta
32     return tempo_prenhez

```

Figura 6

Teve que ser instalado uma biblioteca para trabalhar com o aplicativo Telegram(Figura 7) e inserido o link fornecido pelo BotFather no Telegram (função para criar um Bot), Figura 8.

```

41 import telepot

```

Figura 7


```

43     api="814018385:AAHTC5QAmzxSVn2ozqOSOTLotn5GalWisEQ"
44

```

Figura 8

As figuras 9 e 10, são cadeias de condicionais de uma função que responderão o usuário de acordo com suas respostas e se estiver preparada e em algum período, o Bot enviará uma mensagem com as informações sobre a nutrição que deve ser fornecida.

```

def receber(msg):
    n_msg=msg['message_id']
    text=msg['text']
    _id=msg['from']['id']
    ft=[26,17,12018]
    necessidades=[]
    t=0
    periodo = 0
    if "a" in verificação and "b" in verificação:
        tele.sendMessage(_id,"Erro, digite '/start' para reiniciar")
        for c in range(len(verificação)):
            verificação.pop()
    if text=="/start":
        tele.sendMessage(_id,"Bem vindo ao programa de manejo de gado leiteiro! Escolha uma das opções e escolha sua letra correspondente")
        for c in range(len(verificação)):
            verificação.pop()
    if text=="a" or text=="A":
        tele.sendMessage(_id,"Sua vaca possui 10 meses ou mais?(sim ou não)")
        verificação.append(text)
    if text=="não" and verificação[0]=="a":
        tele.sendMessage(_id,"Seu animal ainda não alcançou a puberdade")
        tele.sendMessage(_id,"aperte aqui '/start' para reiniciar o programa")
    if text=="sim" and verificação[0]=="a":
        tele.sendMessage(_id,"Qual é o peso do animal?(kg):")
        verificação.append(text)
    if numero_vdd(text)==True and verificação[0]=="a":
        text=int(text)
        if text<350:
            tele.sendMessage(_id,"Sua vaca alcançou a puberdade, mas zootecnicamente não está preparada para a monta")
            verificação.append(text)
            tele.sendMessage(_id,"aperte aqui '/start' para reiniciar o programa")
        elif text>=350:

```

Figura 9

```

verificação.append("!")
if len(verificação)==3 and verificação[0]=="b":
    verificação.append(text)
if len(verificação)==4 and numero_vdd(text)==False and verificação[0]=="b":
    t=text.split(" ")
    verificação.append(text)
if len(verificação)==5 and verificação[0]=="b":
    if dias_gestação(t)<100:
        for c in range(len(inicio_lac)):
            necessidades.append(inicio_lac[c])
        periodo="inicio da lactação"
    elif dias_gestação(t)>=100 and dias_gestação(t)<200:
        for s in range(len(meio_lac)):
            necessidades.append(meio_lac[s])
        periodo = "meio da lactação"
    elif dias_gestação(t)>=200 and dias_gestação(t)<220:
        for g in range(len(final_gest)):
            necessidades.append(final_gest[g])
        periodo = "final da lactação"
    elif dias_gestação(t)==220 and dias_gestação(t)<260:
        for h in range(len(vacas_secas)):
            necessidades.append(vacas_secas[h])
        periodo = "período de secagem"
    elif dias_gestação(t)>=260:
        for h in range(len(vacas_secas)):
            necessidades.append(vacas_secas[h])
        periodo="a gestação deveria ter terminado"
    peso_soja,peso_milho=calcula_racao(necessidades,verificação[1])
    tele.sendMessage(_id,f"Sua vaca esta no {periodo}. Se recomenda para esse animal {peso_soja}
    tele.sendMessage(_id,"aperte aqui '/start' para reiniciar o programa")
tele=telepot.Bot(api)
tele.message_loop((receber))

```

Figura 10

No bot é possível ser acessado procurando “MANEJO DE GADO DE LEITE”

Dentro do telegrama será como em uma conversa para familiarizar o produtor ou usuário a tecnologia.

5-Resultados obtidos com a implementação:

Com o Bot no aplicativo é possível ter exatamente as quantidades adequadas de cada nutriente e grão e até mesmo saber se a vaca já está preparada para receber a monta, devido a sua idade e estrutura zootécnica.

6-Dificuldades encontradas durante o trabalho:

Estava difícil de rodar corretamente, estava dando erros no programa e para solucionar foi pesquisado um fórum na internet para auxiliar no Python, outra questão que foi colocado em pauta pela dupla é a dificuldade de achar uma biblioteca adequada para o programa rodar no Telegram (a parte de mensagens em si).

7-Sugestões de trabalhos para serem aplicados na disciplina:

Aparelhos de medição de temperatura, quantidades de chuvas, período de melhor eficiência da planta, nutrição para gado de corte, manejo na avicultura ou suinocultura ou equinocultura, esses exemplos desenvolvidos em Python e voltado para melhorar as atividades no curso de Agronomia.